MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Madhav Institute Of Technology & Science
Gwalior (M.P.)

HOUSE PRICE PREDICTION USING PYTHON PROGRAMMING

Submitted By:

VEDANT KHANDELWAL

(0901CS201134)

Faculty Mentor:

Dr. RANJEET KUMAR SINGH

Submitted to:
DEPARTMENT OF COMPUTER SCIENCE
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR – 474005
JAN-JUNE 2022

Index

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that VEDANT KHANDELWAL (0901CS201134) have submitted the project report titled **–HOUSE PRICE PREDICTION USING PYTHON PROGRAMMING‖** under the mentorship of **Dr. Ranjeet Kumar Singh**, in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

Dr. Ranjeet Kumar Singh
Faculty Mentor
Assistant Professor, CSE

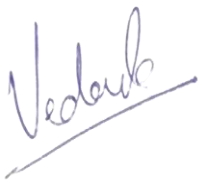
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

I hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship of Dr. **Ranjeet Kumar Singh**, CSE

I declare that I have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.



VEDANT KHANDELWAL

0901CS201134, 2ND Year CSE

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering, for allowing** me to explore this project. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Dr. Ranjeet Kumar Singh, Assistant Professor,** CSE, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

VEDANT KHANDELWAL

0901CS201134, 2ND Year, CSE

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ABSTRACT

HOUSE PRICE PREDICTION MODEL is a model which is useful for getting a approximation of the price of house before buying it. It is based on PYTHON PROGRAMMING and is easily accessible for the users.

House prices increase every year, so there is a need for a system to predict house prices in the future. House price prediction can help the developer determine the selling price of a house and can help the customer to arrange the right time to purchase a house. There are three factors that influence the price of a house which include physical conditions, concept and location. This research aims to predict house prices based on NJOP houses in a city with regression analysis and particle swarm optimization (PSO). PSO is used for selection of affect variables and regression analysis is used to determine the optimal coefficient in prediction.

INTRODUCTION

Machine learning is a subfield of Artificial Intelligence (AI) that works with algorithms and technologies to extract useful information from data. Machine learning methods are appropriate in big data since attempting to manually process vast volumes of data would be impossible without the support of machines. Machine learning in computer science attempts to solve problems algorithmically rather than purely mathematically. Therefore, it is based on creating algorithms that permit the machine to learn. However, there are two general groups in machine learning which are supervised and unsupervised. Supervised is where the program gets trained on pre-determined set to be able to predict when a new data is given. Unsupervised is where the program tries to find the relationship and the hidden pattern between the data [1]. Several Machine Learning algorithms are used to solve problems in the real world today. However, some of them give better performance in certain circumstances, as stated in the No Free Lunch Theorem [2]. Thus, this thesis attempts to use regression algorithms and artificial neural network (ANN) to compare their performance when it comes to predicting values of a given dataset. The performance will be measured upon predicting house prices since the prediction in many regression algorithms relies not only on a specific feature but on an unknown number of attributes that result in the value to be predicted. House prices depend on an individual house specification. Houses have a variant number of features that may not have the same cost due to its location. For instance, a big house may have a higher price if it is located in desirable rich area than being placed in a poor neighbourhood. The data used in the experiment will be handled by using a combination of pre-processing methods to improve the prediction accuracy. In addition, some factors will be added to the local dataset in order to study the relationship between these factors and the sale price in King country , USA

Why we choose House price prediction ?

Problems faced during buying a house:

1)Buying a house is a stressful thing.

2) Buyers are generally not aware of factors that influence the house prices.

3) Many problems are faced during buying a house.

4) Hence real estate agents are trusted with the communication between buyers and sellers as well as laying down a legal contract for the transfer. This just creates a middle man and increases the cost of houses.

OBJECTIVE

This study aims to analyse the accuracy of predicting house prices when using Multiple linear, KNN and Artificial neural network (ANN). Thus, the purpose of this study is to deepen the knowledge in regression methods in machine learning. In addition, the given datasets should be processed to enhance performance, which is accomplished by identifying the necessary features by applying one of the selection methods 2 to eliminate the unwanted variables since each house has its unique features that help to estimate its price. These features may or may not be shared with all houses, which means they do not have the same influence on the house pricing resulting in inaccurate output.

Tools Used

- Python
- Jupyter Notebook
- Tensorflow

Hardware & Software Required

Hardware Requirements

- Processor – (minimum)i3
- Hard Disk – 2 GB
- Memory – 1GB RAM

Software Requirements

- Windows 7(ultimate, enterprise)
- Visual studio (Latest)
- Python
- Jupyter

Methodology /flowchart

We have found out the accurate predictions of the houses/ properties present in the USA for the next upcoming years. Here is the step-by-step process involved

1.Data collection – We have used the dataset of houses of King Country, USA to make our supervised learning model.

2. Data Cleaning

3. Feature Engineering

4. Dimensionality Reduction

5. Splitting the dataset into train , test in 7:3

6.Building a model

7. Calculating the model accuracy using test dataset.

About Dataset

This dataset contains house sale prices for King County, which includes Seattle. It includes homes sold between May 2014 and May 2015.

It's a great dataset for evaluating simple regression models.

First 5 Records of our dataset:-

Date House was Sold	Sale Price	No of Bedrooms	No of Bathrooms	Flat Area (in Sqft)	Lot Area (in Sqft)	No of Floors	Waterfront View	No of Times Visited	...	Overall Grade	Area of the House from Basement (in Sqft)	Basement Area (in Sqft)	Age of House (in Years)	Renovated Year	Zipcode	Latitude	Longitude	Living Area after Renovation (in Sqft)	Lot Area after Renovation (in Sqft)
14 October 2017	221900.0	3	1.00	1180.0	5650.0	1.0	No	None	...	7	1180.0	0	63	0	98178.0	47.5112	-122.257	1340.0	5650
14 December 2017	538000.0	3	2.25	2570.0	7242.0	2.0	No	None	...	7	2170.0	400	67	1991	98125.0	47.7210	-122.319	1690.0	7639
15 February 2016	180000.0	2	1.00	770.0	10000.0	1.0	No	None	...	6	770.0	0	85	0	98028.0	47.7379	-122.233	2720.0	8062
14 December 2017	604000.0	4	3.00	1960.0	5000.0	1.0	No	None	...	7	1050.0	910	53	0	98136.0	47.5208	-122.393	1360.0	5000
15 February 2016	510000.0	3	2.00	1680.0	8080.0	1.0	No	None	...	8	1680.0	0	31	0	98074.0	47.6168	-122.045	1800.0	7503

Information about the dataset , what kind of data types are your variables.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21613 entries, 0 to 21612
Data columns (total 21 columns):
 #    Column                                    Non-Null Count  Dtype
---   ------                                    --------------  -----
 0    ID                                        21613 non-null  int64
 1    Date House was Sold                       21613 non-null  object
 2    Sale Price                                21609 non-null  float64
 3    No of Bedrooms                            21613 non-null  int64
 4    No of Bathrooms                           21609 non-null  float64
 5    Flat Area (in Sqft)                       21604 non-null  float64
 6    Lot Area (in Sqft)                        21604 non-null  float64
 7    No of Floors                              21613 non-null  float64
 8    Waterfront View                           21613 non-null  object
 9    No of Times Visited                       21613 non-null  object
 10   Condition of the House                    21613 non-null  object
 11   Overall Grade                             21613 non-null  int64
 12   Area of the House from Basement (in Sqft) 21610 non-null  float64
 13   Basement Area (in Sqft)                   21613 non-null  int64
 14   Age of House (in Years)                   21613 non-null  int64
 15   Renovated Year                            21613 non-null  int64
 16   Zipcode                                   21612 non-null  float64
 17   Latitude                                  21612 non-null  float64
 18   Longitude                                 21612 non-null  float64
 19   Living Area after Renovation (in Sqft)    21612 non-null  float64
 20   Lot Area after Renovation (in Sqft)       21613 non-null  int64
```

Data cleaning

Detection and Correction of Outliers

Outliers are noisy data that they do have abnormal behaviour comparing with the rest of the data in the same dataset. Outliers can influence the prediction model and performance due to its oddity. There are three types of outliers, which are point, contextual, and collective outliers. Point outlier is an individual data instance that can be considered as odd with respect to 15 the rest of the data. The contextual outlier is an instance of data that can be regarded as odd in a specific context but not otherwise. An example of contextual is the longitude of a location. A collective outlier is a collection of related data instances that can be considered as abnormal with respect to the entire dataset. In supervised, the detection of outliers can be accomplished visually, where a predictive model is built for normal against outliers' classes. Outliers are treated by calculating upper limit and lower limit and replacing the values greater than upper limit with upper limit and values lesser the lower limit with lower limit.

Upper limit =q3+1.5iqr

Lower limit =q1-1.5iqr

Missing values treatment

The problem of missing value is quite common in many real-life datasets. Missing value can bias the results of the machine learning models and/or reduce the accuracy of the model. Missing data is defined as the values or data that is not stored (or not present) for some variable/s in the given dataset. In the dataset, blank shows the missing values In Pandas, usually, missing values are represented by **NaN** . It stands for **Not a Number**.

Why Is Data Missing from The Dataset?

There can be multiple reasons why certain values are missing from the data Reasons for the missing data from the dataset affect the approach of handling missing data. So, it’s necessary to understand why the data could be missing.

Some of the reasons are listed below:

- Past data might get corrupted due to improper maintenance.
- Observations are not recorded for certain fields due to some reasons. There might be a failure in recording the values due to human error.
- The user has not provided the values intentionally.

Why Do We Need To Care About Handling Missing Value?

It is important to handle the missing values appropriately.

- Many machine learning algorithms fail if the dataset contains missing values. However, algorithms like K-nearest and Naive Bayes support data with missing values.
- You may end up building a biased machine learning model which will lead to incorrect results if the missing values are not handled properly.
- Missing data can lead to a lack of precision in the statistical analysis.

Figure Out How To Handle The Missing Data

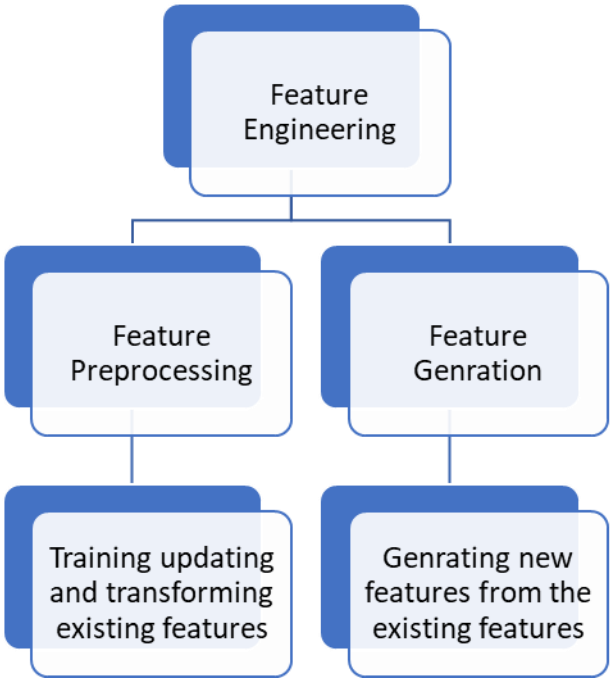
Analyze each column with missing values carefully to understand the reasons behind the missing values as it is crucial to find out the strategy for handling the missing values.
There are 2 primary ways of handling missing values:

1. Deleting the Missing values- Deletion is generally done Dependent variables it is avoided as by using deletion loss of data occurs As in deletion we delete the record with missing value .
2. Imputing the Missing Values- In imputing missing value is replaced with another value like mean, median ,absolute value , most frequent(mode) in our model we used mean in case of numerical variables and most frequent in case of categorical variables.

Feature Engineering

Feature engineering refers to **manipulation — addition, deletion, combination, mutation — of your data set to improve machine learning**

model training, leading to better performance and greater accuracy. Effective feature engineering is based on sound knowledge of the business problem and the available data sources.



By using the variable the variable ‘Renovated Year’ we have derived 2 new variables :- ‘ Ever Renovated’- If the house was renovated then its value will be 1 else 0.

‘Year Since renovated’-if the house was renovated then how many year before the purchase year it was renovated.

```
# deriving two variables from Renovated Year :- Ever Renovated,Year Since Renovated
data['Ever Renovated']=np.where(data['Renovated Year']==0,'No','Yes')
```

```
data['Purchase Year']=pd.DatetimeIndex(data['Date House was Sold']).year
```

```
data['Year Since Renovated']=np.where(data['Ever Renovated']=='Yes',abs(data['Purchase Year']-data['Renovated Year']),0)
```

Dimension Reduction

Dimensionality reduction is a machine learning (ML) or statistical technique of reducing the amount of random variables in a problem by obtaining a set of principal variables. This process can be carried out using a number of methods that simplify the modelling of complex problems, eliminate redundancy and reduce the possibility of the model overfitting and thereby including results that do not belong.

The process of dimensionality reduction is divided into two components, feature selection and feature extraction. In feature selection, smaller subsets of features are chosen from a set of many dimensional data to represent the model by filtering, wrapping or embedding. Feature extraction reduces the number of dimensions in a dataset in order to model variables and perform component analysis.

Methods of dimensionality reduction include:

- Factor Analysis
- Low Variance Filter
- High Correlation Filter
- Backward Feature Elimination
- Forward Feature Selection
- Principal Component Analysis (PCA)
- Linear Discriminant Analysis
- Methods Based on Projections
- t-Distributed Stochastic Neighbour Embedding (t-SNE)
- UMAP
- Independent Component Analysis
- Missing Value Ratio

- Random Forest

Dimensionality reduction is advantageous to AI developers or data professionals working with massive datasets, performing data visualization and analysing complex data. It aids in the process of data compression, allowing the data to take up less storage space as well as reduces computation times.

In our project we first tried to reduce independent variables by calculating correlation between the variables and we got 32 pairs in which correlation was greater then 0.5 and we got 16 variables being practical we cannot remove 16 variables then we used multicollinearity method to remove multicollinearity in which we got 7 variables with VIF>5 then we created a function in which after removing max VIF valued factor we calculated VIF for all if then also the variables VIF value greater than 5 then again remove the factor with max VIF continue till VIF all variables become lesser then 5. By using this we have removed 3 factors

```
X.corr()
```

	No of Bedrooms	No of Bathrooms	Flat Area (in Sqft)	Lot Area (in Sqft)	No of Floors	No of Times Visited	Overall Grade	Area of the House from Basement (in Sqft)	Basement Area (in Sqft)	Age of House (in Years)	...	Ever_Renov
No of Bedrooms	1.000000	0.515818	0.576620	0.031635	0.175536	0.079575	0.349223	0.477550	0.303294	-0.154113	...	
No of Bathrooms	0.515818	1.000000	0.754570	0.087742	0.500762	0.187802	0.635636	0.685112	0.283789	-0.505935	...	
Flat Area (in Sqft)	0.576620	0.754570	1.000000	0.172713	0.354154	0.284656	0.705733	0.876257	0.435139	-0.318196	...	
Lot Area (in Sqft)	0.031635	0.087742	0.172713	1.000000	-0.005176	0.074689	0.102333	0.183483	0.015264	-0.053102	...	
No of Floors	0.175536	0.500762	0.354154	-0.005176	1.000000	0.029504	0.461368	0.524022	-0.245572	-0.489244	...	
No of Times Visited	0.079575	0.187802	0.284656	0.074689	0.029504	1.000000	0.223661	0.167834	0.276974	0.053395	...	
Overall Grade	0.349223	0.635636	0.705733	0.102333	0.461368	0.223661	1.000000	0.705182	0.145232	-0.456711	...	
Area of the House from Basement (in Sqft)	0.477550	0.685112	0.876257	0.183483	0.524022	0.167834	0.705182	1.000000	-0.051804	-0.423859	...	
Basement Area (in Sqft)	0.303294	0.283789	0.435139	0.015264	-0.245572	0.276974	0.145232	-0.051804	1.000000	0.133072	...	
Age of House (in Years)	-0.154113	-0.505935	-0.318196	-0.053102	-0.489244	0.053395	-0.456711	-0.423859	0.133072	1.000000	...	
Latitude	-0.008703	0.024581	0.052547	-0.085674	0.049696	0.006163	0.111228	-0.000787	0.110453	0.148075	...	
Longitude	0.129573	0.223188	0.240094	0.229440	0.125615	-0.078455	0.201733	0.343771	-0.144815	-0.409509	...	
Living Area after Renovation (in Sqft)	0.391768	0.568562	0.756195	0.144509	0.280115	0.280450	0.681366	0.732014	0.200296	-0.326321	...	
Lot Area after Renovation (in Sqft)	0.029264	0.087230	0.183211	0.718526	-0.011204	0.072561	0.107581	0.194100	0.017263	-0.071016	...	
Year Since Renovated	-0.007198	0.003557	0.023491	0.013941	-0.000901	0.093546	-0.024388	0.010484	0.029158	0.203375	...	
Condition_of_the_House_Excellent	0.028148	-0.034271	-0.018172	-0.014494	-0.120524	0.034392	-0.082628	-0.088388	0.127876	0.244330	...	
Condition_of_the_House_Fair	0.004778	0.190434	0.102631	-0.011366	0.317934	-0.037127	0.197510	0.194520	-0.151347	-0.391693	...	
Condition_of_the_House_Good	-0.008847	-0.166036	-0.084002	0.013064	-0.257680	0.022690	-0.140113	-0.142482	0.092539	0.257392	...	
Condition_of_the_House_Okay	-0.051957	-0.077416	-0.065341	0.037612	-0.055951	-0.018557	-0.090561	-0.058935	-0.025312	0.067269	...	
Waterfront_View_Yes	-0.006578	0.063764	0.103835	0.021599	0.023719	0.401856	0.070332	0.072096	0.080595	0.026149	...	
Ever_Renovated_Yes	0.018573	0.050289	0.055095	0.007803	0.006297	0.104051	0.010010	0.023194	0.070969	0.225182	...	
Zipcode_Group_Zipcode_Group_1	-0.010603	-0.032799	-0.058762	0.023658	-0.003385	-0.065000	-0.075495	-0.028362	-0.069149	-0.070111	...	
Zipcode_Group_Zipcode_Group_2	-0.039342	-0.081465	-0.063020	0.052088	-0.067904	0.004754	-0.121379	-0.052404	-0.032302	0.022094	...	
Zipcode_Group_Zipcode_Group_3	-0.074129	-0.034468	-0.078770	-0.041121	0.079211	0.005905	-0.047869	-0.090608	0.006247	0.095882	...	
Zipcode_Group_Zipcode_Group_4	0.024433	0.084049	0.086125	-0.011979	0.071786	0.003509	0.151245	0.086622	0.016422	-0.056974	...	
Zipcode_Group_Zipcode_Group_5	0.019420	0.052771	0.076004	0.015303	0.009203	0.024801	0.095613	0.062322	0.040662	-0.009965	...	
Zipcode_Group_Zipcode_Group_6	0.090177	0.123266	0.160024	-0.023260	0.069857	0.068144	0.200548	0.129299	0.090206	0.025718	...	
Zipcode_Group_Zipcode_Group_7	0.016725	0.037750	0.051203	-0.027428	0.064981	-0.012548	0.077126	0.029473	0.051128	0.101486	...	
Zipcode_Group_Zipcode_Group_8	0.102736	0.110018	0.169564	-0.006987	-0.008633	0.065335	0.156952	0.134015	0.101159	-0.002705	...	
Zipcode_Group_Zipcode_Group_9	0.035694	0.067872	0.090250	0.002667	0.005868	0.012923	0.048638	0.087396	0.023775	-0.002492	...	

```
# Importing Variance_inflation_Factor funtion from the Statsmodels
from statsmodels.stats.outliers_influence import variance_inflation_factor

vif_data = X

## Calculating VIF for every column
VIF = pd.Series([variance_inflation_factor(vif_data.values, i) for i in range(vif_data.shape[1])], index = vif_data.columns)
VIF
```

```
No of Bedrooms                              1.639469
No of Bathrooms                             3.374716
Flat Area (in Sqft)                      1562.844886
Lot Area (in Sqft)                          2.108068
No of Floors                                2.127712
No of Times Visited                         1.432393
Overall Grade                               2.967508
Area of the House from Basement (in Sqft)1271.671615
Basement Area (in Sqft)                   364.360177
Age of House (in Years)                     2.629408
Latitude                                    2.471395
Longitude                                   1.672677
Living Area after Renovation (in Sqft)      3.068245
Lot Area after Renovation (in Sqft)         2.144333
Year Since Renovated                        2.788105
Condition_of_the_House_Excellent           53.578090
Condition_of_the_House_Fair               166.129924
Condition_of_the_House_Good               141.324489
Condition_of_the_House_Okay                 6.703057
Waterfront_View_Yes                         1.208442
Ever_Renovated_Yes                          2.955788
Zipcode_Group_Zipcode_Group_1               1.538236
Zipcode_Group_Zipcode_Group_2               2.570589
Zipcode_Group_Zipcode_Group_3               2.818544
Zipcode_Group_Zipcode_Group_4               3.193000
Zipcode_Group_Zipcode_Group_5               1.728270
Zipcode_Group_Zipcode_Group_6               2.014956
Zipcode_Group_Zipcode_Group_7               1.233750
Zipcode_Group_Zipcode_Group_8               1.389636
Zipcode_Group_Zipcode_Group_9               1.048576
dtype: float64
```

After performing dimension reduction variables left and there VIF value:-

```
No of Bedrooms                                1.638996
No of Bathrooms                               3.373697
Lot Area (in Sqft)                            2.107484
No of Floors                                  2.127632
No of Times Visited                           1.432369
Overall Grade                                 2.957083
Area of the House from Basement (in Sqft)     4.580409
Basement Area (in Sqft)                       1.974965
Age of House (in Years)                       2.626375
Latitude                                      2.471300
Longitude                                     1.672653
Living Area after Renovation (in Sqft)        3.063985
Lot Area after Renovation (in Sqft)           2.144054
Year Since Renovated                          2.788073
Condition_of_the_House_Excellent              1.206482
Condition_of_the_House_Good                   1.251486
Condition_of_the_House_Okay                   1.025385
Waterfront_View_Yes                           1.208288
Ever_Renovated_Yes                            2.955542
Zipcode_Group_Zipcode_Group_1                 1.538208
Zipcode_Group_Zipcode_Group_2                 2.570566
Zipcode_Group_Zipcode_Group_3                 2.818465
Zipcode_Group_Zipcode_Group_4                 3.192396
Zipcode_Group_Zipcode_Group_5                 1.728021
Zipcode_Group_Zipcode_Group_6                 2.014722
Zipcode_Group_Zipcode_Group_7                 1.233623
Zipcode_Group_Zipcode_Group_8                 1.389344
Zipcode_Group_Zipcode_Group_9                 1.048569
dtype: float64
```

Building a Model

In Our Project we have use three different models

Multiple Linear Regression:- Multiple Linear Regression (MLR) is a supervised technique used to estimate the relationship between one dependent variable and more than one independent variables. Identifying the correlation and its cause-effect helps to make predictions by using these relations [4]. To estimate these relationships, the prediction accuracy of the model is essential; the complexity of the model is of more interest. However, Multiple Linear Regression is prone to many problems such as multicollinearity, noises, and overfitting, which effect on the prediction accuracy. Regularised regression plays a significant part in Multiple Linear Regression because it helps to reduce variance at the cost of introducing some bias, avoid the overfitting problem and solve ordinary least squares (OLS) problems. There are two types of regularisation techniques L1 norm (least absolute deviations) and L2 norm (least squares). L1 and L2 have different cost functions regarding model complexity [5].

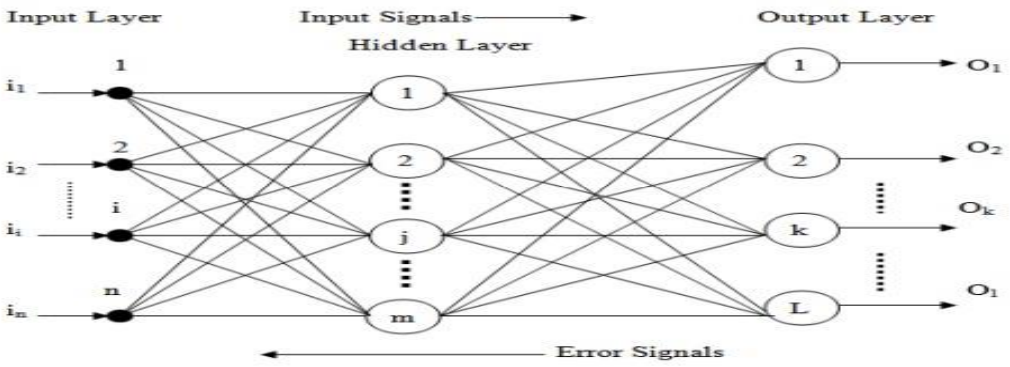
K Neighbours Regressor :- KNN regression is a non-parametric method that, in an intuitive manner, approximates the association between independent variables and the continuous

outcome by averaging the observations in the same *neighbourhood*. The size of the neighbourhood needs to be set by the analyst or can be chosen using cross-validation (we will see this later) to select the size that minimises the mean-squared error.

While the method is quite appealing, it quickly becomes impractical when the dimension increases, i.e., when there are many independent variables.

Artificial Neural Network (ANN): - Artificial neural network (ANN) is an attempt to simulate the work of a biological brain. The brain learns and evolves through the experiments that it faces through time to make decisions and predict the result of particular actions. Thus, ANN tries to simulate the brain to learn the pattern in a given data to predict the output of that data whether the expected data was provided in the learning process or not.

ANN is based on an assemblage of connected elements or nodes called neurons. Neurons act as channels that take an input, process it, and then pass it to other neurons for further processing. This transaction or the process of transferring data between neurons is handled in layers. Layers consist of at least three layers, input layer, one or more of hidden layers and output layer. Each layer holds a set of neurons that takes input and process data and finally pass the output to other neurons in the next layer. This process is repetitive until the output layer has been reached, so 7 eventually, the result can be presented. ANN architecture is shown in the following figure as is also known as feed-forward, which values pass in one direction.



ANN architecture The data that is being held in each neuron is called activation. Activation value ranges from 0 to 1. As shown in figure , each neuron is linked to all neurons in the previous layer. Together, all activations from the first layer will decide if the activation will be

triggered or not, which is done by taking all activations from the first layer and compute their weighted sum . $w1a1 + w2a2 + w3a3 +$

$\cdots + wnan$ However, the output could be any number when it should be only between 0 and 1. Thus, specifying the range of the output value to be within the accepted range. It can be done by using the Sigmoid function that will put the output to be ranging from 0 to 1. Then the bias is added for inactivity to the equation so it can limit the activation to when it is meaningfully active. $\sigma(w1a1 + w2a2 + w3a3 + \cdots + wnan - b)$ Where a is activation, w presents the weight, b is the bias and σ is the sigmoid function. Nevertheless, after getting the final activation, its predicted value needs to be compared with the actual value. The difference between these values is considered as an error, and it is calculated with the cost function. The cost function helps to detect the error percentage in the model, which needs to be reduced. Applying back-propagation on the model reduces the error percentage by running the procedures backwards to check on how the weight and bias are affecting the cost function. 8 Back-propagation is simply the process of reversing the whole activations transference among neurons. The method calculates the gradient of the cost function concerning the weight. It is performed in the training stage of the feed-forward for supervised learning.

RESULT

We have predicted the House Price using three different ML model algorithms.

The score of our Multiple Linear Regression is around 84%and for KNN is around 84%,so this model had room for improvement. Then we got an accuracy of approx. 88 % with keras regression model.

Using Linear Regression:-

```
Variance Regression Score:  0.8472020433417637
```

Using KNN:-

```
Variance Regression Score:  0.8421952141914069
```

Using ANN:-

```
Variance Regression Score:  0.8792841007706511
```

CONCLUSION

The main aim of this project is to determine the prediction of prices. In this paper, we have discovered many algorithms and application of machine learning techniques with the objective of buying the real estate properties and to predict the worth in the future of the owned real estate properties. Price can be predicted through many factors like the surrounding, marketplaces and many related factors with the house.We have first cleaning and exploring of the input data. We have performed ensembles of regression k-nearest neighbours, multi-linear regression and Artificial neural network ANN.

SCOPE OF THE PROJECT

Future work on this study could be divided into seven main areas to improve the result even further. Which can be done by:

- The used pre-processing methods do help in the prediction accuracy. However, experimenting with different combinations of pre-processing methods to achieve better prediction accuracy.
- Make use of the available features and if they could be combined as binning features has shown that the data got improved.

-House Price prediction, is important to drive Real Estate efficiency. As earlier, House prices were determined by calculating the acquiring and selling price in a locality. Therefore, the House Price prediction model is very essential in filling the information gap and improve Real Estate efficiency.

BIBLIOGRAPHY

- https://www.kaggle.com/code/laylalayla/predicting-house-prices-using-linear-regression/data
- Stack overflow
- W3School
- https://www.tensorflow.org/api_docs/python/tf
- https://scikit-learn.org/stable/supervised_learning.html#supervised-learning
- Analytic Vidhya