VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590 018.

An Internship Report
On

"HOUSE PRICE PREDICTION"

03.09.2022 to 02.10.2022

Submitted in partial fulfilment of the for the award of the degree of

Bachelor of Engineering

In

Computer Science & Engineering

Submitted by

KOMAL BHAT **1VI19CS046**

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
VEMANA INSTITUTE OF TECHNOLOGY
BENGALURU – 560034
2022-2023

Karnataka Reddy Jana Sangha®

VEMANA INSTITUTE OF TECHNOLOGY

(Affiliated to Visvesvaraya Technological University, Belagavi)

Koramangala, Bengaluru-560034.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the Internship/Professional Practice work entitled "**HOUSE PRICE PREDICTION**" is a bonafide work carried out by **Ms. KOMAL BHAT (1VI19CS046)** during the academic year 2022-23 in partial fulfilment of the requirement for the award of **Bachelor of Engineering in Computer Science & Engineering** of the **Visvesvaraya Technological University, Belagavi.** It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report. The internship report has been approved as it satisfies the academic requirements in respect of the Internship/ Professional Practice prescribed for the said degree.

Guide	**Head of the Department**	**Principal**
Dr Kantharaju H C	Dr. Ramakrishna M	Dr. Vijayasimha Reddy B.G

External Viva

Name of the Examiner	Signature with date
1. _____	_____
2. _____	_____

ACKNOWLEDGEMENT

I sincerely thank Visvesvaraya Technology University for providing a platform to do the internship work.

I express my sincere thanks to **Dr. Vijayasimha Reddy B G**, Principal, Vemana Institute of Technology, Bengaluru, for providing necessary facilities and motivation to carry out internship work successfully.

I express heartfelt gratitude and humble thanks to **Dr. M. Ramakrishna**, Professor and Head, Computer Science and Engineering, Vemana Institute of Technology, for his constant encouragement, inspiration and help to carry out internship work successfully.

I am very thankful to my external guide, **Santosh S,** Data Scientist-Full Stack**,** at Vivarttana Technologies Pvt. Ltd. who has given in-time valuable instructions and put me in contact with experts in the field, with extensive guidance regarding practical issues.

I would like to express my sincere gratitude towards my internal guide, **Dr. Kantharaju H. C,** Associate Professor for providing encouragement and inspiration throughout the internship.

I thank internship coordinators **Prof. Naveen H S** and **Prof. Kavitha Bai A.S** for their cooperation and support during the internship work

I am thankful to all the teaching and non-teaching staff members of Computer Science and Engineering department for their help and much needed support throughout the internship.

Komal Bhat

1VI19CS046


Gyaan
Vivarttana Technologies Pvt. Ltd.
Your Success Our Passion
Internship Completion Certificate
This is to certify that
Mr. / Ms. / Mrs Komal Bhat
bearing USN 1VI19CS046
From Vemana Institute of Technology
successfully completed the Internship
in Python & Machine Learning with Visualization during
September 2022 for the period of 4 Weeks
Authorised Signatory
Date: 30/09/2022

ABSTRACT

Real estate price prediction is crucial for the establishment of real estate policies and can help real estate owners and agents make informative decisions. The aim of this study is to employ actual transaction data and machine learning models to predict prices of real estate. The actual transaction data contain attributes and transaction prices of real estate that respectively serve as independent variables and dependent variables for machine learning models. Numerical results indicated that the least squares support vector regression outperforms the other machine learning model in terms of forecasting accuracy. Furthermore, forecasting results generated by the least squares support vector regression are superior to previous related studies of real estate price prediction in terms of the average absolute percentage error. Thus, the machine learning - based model is a substantial and feasible way to forecast real estate prices, and the least squares linear regression can provide relatively competitive and satisfactory results.

Keywords: Real Estate, Support Vector Regression, Machine Learning, Linear Regression

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In this study, Python programming language with several Python packages will be used. 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, some of them give better performance in certain circumstances. Thus, this thesis attempts to use regression algorithms and artificial neural networks (ANN) to compare their performance when it comes to predicting the values of a given dataset [1].

The performance will be measured in 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 their location. For instance, a big house may have a higher price if it is located in a desirable rich area than being placed in a poor neighborhood. The data used in the experiment will be handled by using a combination of pre-processing methods to improve the prediction accuracy. However, there are two general groups in machine learning which are supervised and unsupervised. Supervised is where the program gets trained on a pre-determined set to be able to predict when new data is given. Unsupervised is where the program tries to find the relationship and the hidden pattern between the data. Several Machine Learning algorithms are used to solve problems in the real world today [2].

Thousands of houses are sold every day. There are some questions every buyer asks himself: What is the actual price that this house deserves? Am I paying a fair price? In this paper, a machine learning model is proposed to predict a house price based on data related to the house (its size, the year it was built, etc.). During the development and evaluation of our model, we will show the code used for each step followed by its output. This will facilitate the reproducibility of our work.

1.1 OBJECTIVE

This project is proposed to predict house prices and to get better and more accurate results. The stacking algorithm is applied to various regression algorithms to see which algorithm has the most accurate and precise results. This would be of great help to the people because house pricing is a topic that concerns a lot of citizens whether rich or middle class as one can never judge or estimate the pricing of a house based on locality or facilities available. To accomplish this task, the python programming language is used. Python is a high-level programming language for general-purpose programming. It enables clear programming on both small and large scales. It is an easily readable language.

CHAPTER 2

ORGANIZATION OVERVIEW

Vivarttana Technologies is an indigenous & innovative bunch! A strange but efficient mixture of what it takes to build a good software package. Vivarttana Technologies is passionate about Software Engineers and committed to providing outstanding services to our students.

Team members of Vivarttana technologies have overall 20+ years of experience in the Software industry and 30+ years of experience in teaching. Vivarttana Technologies know what the industry needs and train the students as per the requirements of the industrial demands.

A team of seasoned professionals worked in Retail, BFSI, Manufacturing & Automobile, Travel, and Logistics vertical / domain across the US, Europe, APAC, and Africa regions. Strong technology expertise in - Data Warehousing, Big-Data, Web Technologies, Java Technology, etc.

The race for digital transformation is on. In this globally connected on-demand world with rapid advancements in internet technologies, businesses worldwide are under constant pressure to add innovative real-time capabilities to their applications to respond to market opportunities. Every business worldwide is building event-driven, real-time applications - from financial services, transportation, and energy, to retail, healthcare, and Gaming companies. We endeavor to make it easy to develop innovative real-time applications and efficient to operate them in production.

Vivarttana team has a proven record of building highly scalable, world-class consulting processes that offer tremendous business advantages to our clients in the form of huge cost benefits, definitive results, and consistent project deliveries across the globe. Prominently strive to improve your business by delivering the full range of competencies including operational performance, developing and applying business strategies to improve.

Vivarttana organization delivered successful projects we pride ourselves on being a sought-after data science analyst. Our organization believes that our customer's success is our passion.

Some of our organization services included are:

Internet of Things – Implementing the present iot (neno) projects that are related to hardware and software devices, and IoT devices facing a problem in the real world, so we tackle the problem and implement it in the real world.

Data Science Analysis – Develop a machine learning model and deep learning models to

predict the data and analyze the data.

Full Stack Development - Develop responsive, functional, and super-fast websites. Keep User Experience in mind while creating websites. A website should load quickly and should be accessible even on a small view-port and slow internet connection.

Android Application Development - Offer a wide range of professional Android, IOS & Hybrid app development services for our global clients, from a start-up to a large enterprise.

Design - Professional Graphic design, Brochure design & Logo design. We are experts in crafting visual content to convey the right message to the customers.

Consultancy - Provide you with expert advice on your design and development requirements.

Vision

Setting a benchmark in Software Application & Product Development and Transforming Individuals to build their Passionate Dream Career.

Mission

Empower every person and every Company to achieve more with our commitment to Employee development and Client partnerships.

Fig 2.1 Logo of the company

In fig 2.1, shows the logo of the company

CHAPTER 3

SYSTEM SPECIFICATION

3.1 Hardware Requirements

- Processor : i3
- RAM : 4GB
- Hard Disk : 40GB
- Processor Speed : 2.4GHZ
- System Type : 64-bit/32-bit operating system

3.2 Software Requirements

- Language : Python
- Operating System : Windows 10

CHAPTER 4

SOFTWARE SPECIFICATION

1. MACHINE LEARNING:

Machine Learning is a system that can learn from example through self-improvement and without being explicitly coded by a programmer. The breakthrough comes with the idea that a machine can singularly learn from the data (i.e., example) to produce accurate results. Machine learning combines data with statistical tools to predict an output. This output is then used by corporate to make actionable insights. Machine learning is closely related to data mining and Bayesian predictive modelling. The machine receives data as input and uses an algorithm to formulate answers. A typical machine learning tasks are to provide a recommendation. For those who have a Netflix account, all recommendations of movies or series are based on the user's historical data. Tech companies are using unsupervised learning to improve the user experience by personalizing recommendations. Machine learning is also used for a variety of tasks like fraud detection, predictive maintenance, portfolio optimization, automated task, and so on.

Machine Learning is subcategorized as:

i. Supervised Learning
ii. Unsupervised Learning
iii. Reinforcement Learning

Supervised Learning: In supervised learning, the computer is provided with example inputs that are labelled with their desired outputs. The purpose of this method is for the algorithm to be able to “learn” by comparing its actual output with the “taught” outputs to find errors and modify the model accordingly. Supervised learning, therefore, uses patterns to predict label values on additional unlabelled data.

For example, with supervised learning, an algorithm may be fed data with images of sharks labelled as fish and images of oceans labeled as water. By being trained on this data, the supervised learning algorithm should be able to later identify unlabelled shark images as fish and unlabelled ocean images as water. A common use case of supervised learning is to use historical data to predict statistically likely future events.

Unsupervised Learning: An unsupervised model in unsupervised learning, data is unlabelled, so the learning algorithm is left to find commonalities among its input data. As unlabelled data are more abundant than labelled data, machine learning methods that facilitate unsupervised learning are particularly valuable.

The goal of unsupervised learning may be as straightforward as discovering hidden patterns within a dataset, but it may also have a goal of feature learning, which allows the computational machine to automatically discover the representations that are needed to classify raw data. Unsupervised learning is commonly used for transactional data. You may have a large dataset of customers and their purchases, but as a human, you will likely not be able to make sense of what similar attributes can be drawn from customer profiles and their types of purchases. With this data fed into an unsupervised learning algorithm, it may be determined that women of a certain age range who buy unscented soaps are likely to be pregnant, and therefore a marketing campaign related to pregnancy and baby products can be targeted to this audience in order to increase their number of purchases.

Without being told a "correct" answer, unsupervised learning methods can look at complex data that is more expansive and seemingly unrelated in order to organize it in potentially meaningful ways. Unsupervised learning is often used for anomaly detection including for fraudulent credit card purchases, and recommender systems that recommend what products to buy next. In unsupervised learning, untagged photos of dogs can be used as input data for the algorithm to find likenesses and classify dog photos together.

The Machine Learning commonly uses the following techniques:

Convolution neural networks: A Convolution neural network (CNN) is a computational model. Information flows through the network that affects the structure of the CNN based on that input and output.

Decision trees: Decision tree is a graph that uses a branching method to illustrate every possible outcome of a decision.

Naïve Bayes' algorithm: Naïve Bayes' classifiers are a family of simple "Probabilistic classifiers" based on applying Bayes' theorem with strong independence assumptions between the features.

Genetic algorithms: Genetic algorithm (GA) is adaptive heuristic search algorithm based on evolutionary ideas of genetics and natural selection.

Nearest neighbor method: The principle behind nearest neighbor method is to find a predefined number of training samples closest in distance to the new point and predict the label from these.

Rule induction: Rule induction is an area of machine learning, where the formal rules are extracted from the set. The extracted rules represent pattern in the data.

To best apply these advanced techniques, they must be integrated with data warehouse as well as interactive business analysis tools. In the first the information is collected from review sites, the source of information or Data Sources can be from product reviews, which is often available in multiple formats like csv, ASCII/text, flat files.

Reinforcement Learning: Reinforcement Learning is the ability of an agent to interact with the environment and find out what is the best outcome. It follows the concept of the hit and trial method. The agent is rewarded or penalized with a point for a correct or a wrong answer, and based on the positive reward points gained the model trains itself. And again, once trained it gets ready to predict the new data presented to it.

Main points in Reinforcement learning –

Input: The input should be an initial state from which the model will start.

Output: There are many possible outputs as there are a variety of solutions to a particular problem.

Training: The training is based upon the input. The model will return a state and the user will decide to reward or punish the model based on its output.

Machine learning Approaches-

Decision tree learning: Decision tree learning uses a decision tree as a predictive model, which maps observations about an item to conclusions about the item's target value. • Association rule learning: Association rule learning is a method for discovering interesting relations between variables in large databases.

Artificial neural networks: An artificial neural network (ANN) learning algorithm, usually called "neural network" (NN), is a learning algorithm that is vaguely inspired by biological neural networks. Computations are structured in terms of an interconnected group of artificial neurons, processing information using a connectionist approach.

Deep learning: Falling hardware prices and the development of GPUs for personal use in the

last few years have contributed to the development of the concept of deep learning which consists of multiple hidden layers in an artificial neural network. This approach tries to model the way the human brain processes light and sound into vision and hearing. Some successful applications of deep learning are computer vision and speech recognition.

Inductive logic programming: Inductive logic programming (ILP) is an approach to rule learning using logic Programming as a uniform representation for input examples, background knowledge, and hypotheses.

2. PYTHON:

Python is a general-purpose high-level programming language. Python was developed by Guido Van Rossum in 1989 while working at National Research Institute in the Netherlands. But officially Python was made available to the public in 1991. The official python date of Birth for Python is Feb 20th, 1991. Python is recommended as the first programming language for beginners. Python is a simple programming language. When we read the Python program, we can feel like reading English statements. The syntaxes are very simple and only 30+ keywords are available. When compared with other languages, we can write programs with a very smaller number of lines. Hence more readability and simplicity. We can reduce the development and cost of the project. Python is a high-level programming language and hence it is a programmer-friendly language. Being a programmer, we are not required to concentrate on low-level activities like memory management and security, etc.

CHAPTER 5

SOURCE CODE

```
#Import libraries
import pandas as pd
import numpy as np import matplotlib
from matplotlib import pyplot as plt
#Load dataset
df1 =pd.read_csv("input/bengaluru-house-price-data/Bengaluru_House_Data.csv")
df1.head()
#Data cleaning
df2.isnull().sum()
df2.shape
df3 = df2.dropna()
df3.isnull().sum()
#Explore total sqft feature
def convert_sqft_to_num(x):
tokens = x.split('-')
if len(tokens) == 2:
return (float(tokens[0]) + float(tokens[1]))/2 try:
return float(x) except:
return None
df4 = df3.copy()
df4.total_sqft = df4.total_sqft.apply(convert_sqft_to_num)
df4 = df4[df4.total_sqft.notnull()]
df4.head(2)
#Examine locations which is a categorical variable. we need to apply the dimensionality reduction technique here to reduce the number of locations
df5.location = df5.location.apply(lambda x: x.strip())
location_stats = df5['location'].value_counts(ascending=False) location_stats
#Dimensionality reductions
#Any location having less than 10 data points should be tagged as" other" location. This way
```

number of categories can be reduced by huge amount.

```
location_stats_less_than_10 = location_stats[location_stats<=10] location_stats_less_than_10
df5.location = df5.location.apply(lambda x: 'other' if x in location_stats_less_than_10 else x)
len(df5.location.unique())
df5.head(10)
#Outlier removal using business logic
df5[df5.total_sqft/df5.bhk<300].head()
df5.shape
df6 = df5[~(df5.total_sqft/df5.bhk<300)]
df6.shape
#Outlier removal using standard deviation and mean
def remove_pps_outliers(df):
df_out = pd.DataFrame()
for key, subdf in df.groupby('location'):
m = np.mean(subdf.price_per_sqft)
 st = np.std(subdf.price_per_sqft)
reduced_df = subdf[(subdf.price_per_sqft>(m-st)) & (subdf.price_per_sqft<=(m+st))]
df_out = pd.concat([df_out,reduced_df],ignore_index=True)
return df_out
df7 = remove_pps_outliers(df6)
df7.shape
#Checking for a given location how does the 2BHK and 3BHK property prices look like
def plot_scatter_chart(df,location):
bhk2 = df[(df.location==location) & (df.bhk==2)]
bhk3 = df[(df.location==location) & (df.bhk==3)]
matplotlib.rcParams['figure.figsize'] = (15,10)
plt.scatter(bhk2.total_sqft,bhk2.price,color='blue',label='2BHK',s=50)
plt.scatter(bhk3.total_sqft,bhk3.price,marker='+',color='green',label='3BHK',s=50)
plt.xlabel("Total Square Feet Area")
plt.ylabel("Price (Lakh Indian Rupees)")
plt.title(location)
```

```
plt.legend()
#Build a model
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2,random_state=10)
 from sklearn.linear_model import LinearRegression
lr_clf = LinearRegression()
lr_clf.fit(X_train,y_train)
lr_clf.score(X_test,y_test)
from sklearn.model_selection import ShuffleSplit
from sklearn.model_selection import cross_val_score
cv=ShuffleSplit(n_splits=5,test_size=0.2,random_state=0)
cross_val_score(LinearRegression(), X, y, cv=cv)
#Test the model for few properties
def predict_price(location,sqft,bath,bhk):
loc_index = np.where(X.columns==location)[0][0]
x = np.zeros(len(X.columns))
x[0] = sqft
x[1] = bath
x[2] = bhk
if loc_index >= 0: x[loc_index] = 1
return lr_clf.predict([x])[0]
predict_price('1st Phase JP Nagar',1000, 3, 3)[0]
#Export the tested model to a pickle model
import pickle
with open('banglore_home_prices_model.pickle','wb') as f:
pickle.dump(lr_clf,f)
#Export location and column information to a file that will be useful later on in our prediction
application
import json columns = {
'data_columns' : [col.lower() for col in X.columns]
}
```

```
#Use one hot encoding for location
dummies = pd.get_dummies(df10.location)
dummies.head(3)
#Now we can remove those 2 bhk apartments whose price_per_sqft is less than the mean price_per_sqft of 1 bhk apartment
def remove_bhk_outliers(df):
exclude_indices = np.array([])
for location, location_df in df.groupby('location'):
bhk_stats = {}
for bhk, bhk_df in location_df.groupby('bhk'):
bhk_stats[bhk] = {
'mean': np.mean(bhk_df.price_per_sqft),'std':
np.std(bhk_df.price_per_sqft), 'count':
bhk_df.shape[0]
}
for bhk, bhk_df in location_df.groupby('bhk'):stats
= bhk_stats.get(bhk-1)
if stats and stats['count']>5:
exclude_indices=np.append(exclude_indices,
bhk_df[bhk_df.price_per_sqft<(stats['mean'])].index.values)
return df.drop(exclude_indices,axis='index')
df8 = remove_bhk_outliers(df7)
# df8 = df7.copy()
df8.shape
plot_scatter_chart(df8,"Rajaji Nagar")
plot_scatter_chart(df8,"Hebbal")
import matplotlib
matplotlib.rcParams["figure.figsize"] = (20,10)
plt.hist(df8.price_per_sqft,rwidth=0.8)
plt.xlabel("Price Per Square Feet")
plt.ylabel("Count")
```

```
#examine locations for dimension reductions
df5.location = df5.location.apply(lambda x: x.strip())
location_stats = df5['location'].value_counts(ascending=False)
location_stats
#finding best modal
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import Lasso
from sklearn.tree import DecisionTreeRegressor

def find_best_model_using_gridsearchcv(X,y):
    algos = {
        'linear_regression' : {
            'model': LinearRegression(),
            'params': {
                'normalize': [True, False]
            }
        },
        'lasso': {
            'model': Lasso(),
            'params': {
                'alpha': [1,2],
                'selection': ['random', 'cyclic']
            }
        },
        'decision_tree': {
            'model': DecisionTreeRegressor(),
            'params': {
                'criterion' : ['mse','friedman_mse'],
                'splitter': ['best','random']
            }
        }
    }
    scores = []
```

```
cv = ShuffleSplit(n_splits=5, test_size=0.2, random_state=0)
for algo_name, config in algos.items():
gs = GridSearchCV(config['model'], config['params'], cv=cv, return_train_score=False)
gs.fit(X,y)
scores.append({ 'model':
algo_name,
'best_score': gs.best_score_,
'best_params': gs.best_params_
})
return pd.DataFrame(scores,columns=['model','best_score','best_params'])
find_best_model_using_gridsearchcv(X,y)
```

CHAPTER 6

RESULTS AND SCREENSHOTS

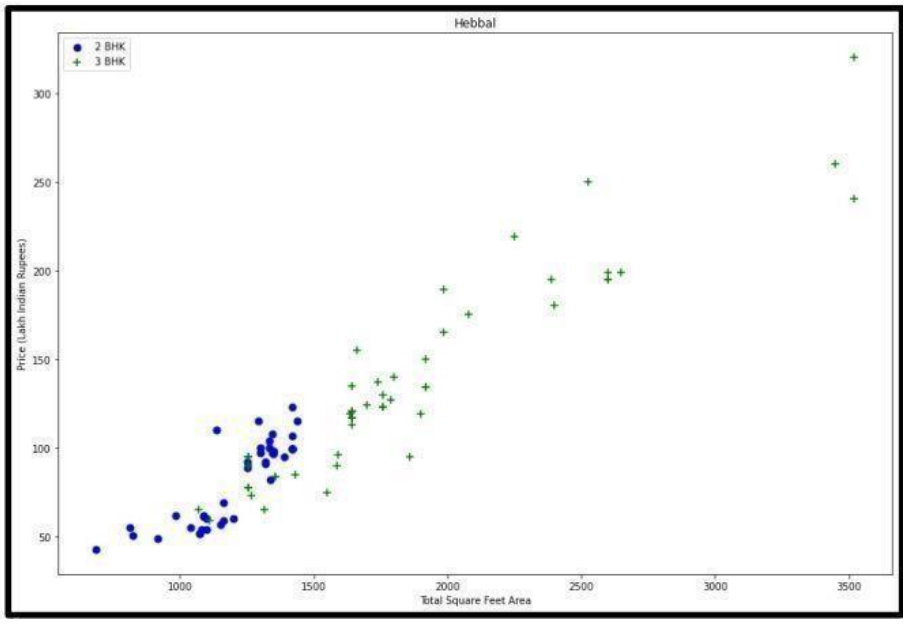


Fig 6.1: Scatter Chart to visualize price_per_sqft

In this Fig 6.1, Scatter chart is used to visualize price_per_sqft for 2BHK and 3BHK Properties.

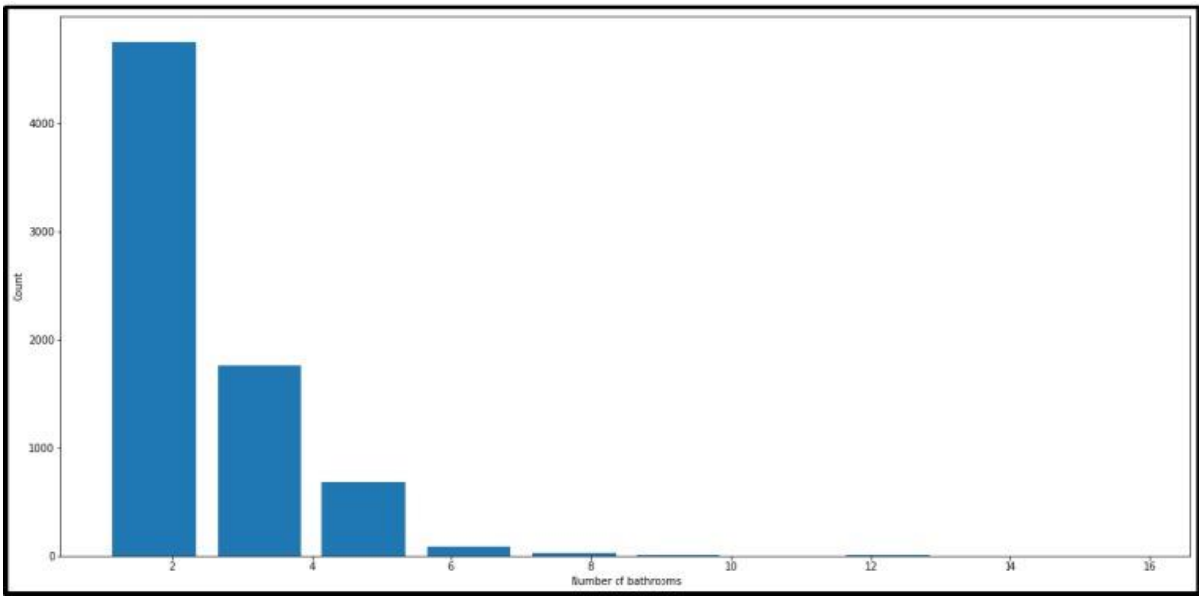


Fig 6.2: Bar Chart to visualize outlier removal

In this Fig 6.2, Using bathroom feature, visualization of outlier removal was plotted.

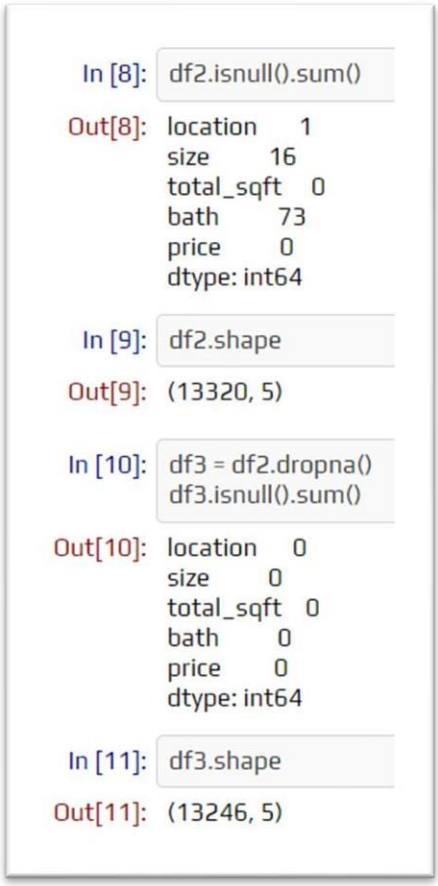


Fig 6.3: Data cleaning

In fig 6.3, isnull() function finds missing values in a series object. It returns a Boolean same-sized object that indicates if the values are NA or not. Missing values are translated to True, whereas non-missing values are assigned to False.

Fig 6.4: Feature Engineering

In fig 6.4, Feature engineering is the process of creating new features from existing data, which is frequently dispersed over many linked tables. Feature engineering is collecting important information from data and consolidating it into a single table that can subsequently be utilized to train a machine learning model.

```
In [13]: def is_float(x):
             try:
                 float(x)
             except:
                 return False
             return True

In [14]: 2+3
Out[14]: 5

In [15]: df3[~df3['total_sqft'].apply(is_float)].head(10)
```

Out[15]:

	location	size	total_sqft	bath	price	bhk
30	Yelahanka	4 BHK	2100 - 2850	4.0	186.000	4
122	Hebbal	4 BHK	3067 - 8156	4.0	477.000	4
137	8th Phase JP Nagar	2 BHK	1042 - 1105	2.0	54.005	2
165	Sarjapur	2 BHK	1145 - 1340	2.0	43.490	2
188	KR Puram	2 BHK	1015 - 1540	2.0	56.800	2
410	Kengeri	1 BHK	34.46Sq. Meter	1.0	18.500	1
549	Hennur Road	2 BHK	1195 - 1440	2.0	63.770	2
648	Arekere	9 Bedroom	4125Perch	9.0	265.000	9
661	Yelahanka	2 BHK	1120 - 1145	2.0	48.130	2
672	Bettahalsoor	4 Bedroom	3090 - 5002	4.0	445.000	4

Fig 6.5 :Total sqft range

In fig 6.5, shows that total_sqft can be a range (e.g. 2100-2850). For such a case, we can just take an average of min and max value in the range.

```
In [20]: df5 = df4.copy()
         df5['price_per_sqft'] = df5['price']*100000/df5['total_sqft']
         df5.head()
```

Out[20]:

	location	size	total_sqft	bath	price	bhk	price_per_sqft
0	Electronic City Phase II	2 BHK	1056.0	2.0	39.07	2	3699.810606
1	Chikka Tirupathi	4 Bedroom	2600.0	5.0	120.00	4	4615.384615
2	Uttarahalli	3 BHK	1440.0	2.0	62.00	3	4305.555556
3	Lingadheeranahalli	3 BHK	1521.0	3.0	95.00	3	6245.890861
4	Kothanur	2 BHK	1200.0	2.0	51.00	2	4250.000000

```
In [21]: df5_stats = df5['price_per_sqft'].describe()
         df5_stats
Out[21]: count    1.320000e+04
         mean     7.920759e+03
         std      1.067272e+05
         min      2.678298e+02
         25%      4.267701e+03
         50%      5.438331e+03
         75%      7.317073e+03
         max      1.200000e+07
         Name: price_per_sqft, dtype: float64

In [22]: df5.to_csv("bhp.csv",index=False)
```

Fig 6.6:Categorical Variables

In fig 6.6, shows the locations that are categorical variables so that it need to apply the dimensionality reduction technique here to reduce the number of locations.

```
In [28]: location_stats_less_than_10 = location_stats[location_stats<=10]
         location_stats_less_than_10

Out[28]: Kalkere                10
         Nagadevanahalli        10
         Ganga Nagar            10
         Sector 1 HSR Layout    10
         Basapura               10
                                ..
         Saptagiri Layout        1
         Jakkasandra             1
         Bapuji Layout           1
         anjananager magdi road  1
         RK Colony               1
         Name: location, Length: 1047, dtype: int64

In [29]: len(df5.location.unique())

Out[29]: 1287

In [30]: df5.location = df5.location.apply(lambda x: 'other' if x in location_stats_less_than_10 else x)
         len(df5.location.unique())

Out[30]: 241

In [31]: df5.head(10)
```

Out[31]:

	location	size	total_sqft	bath	price	bhk	price_per_sqft
0	Electronic City Phase II	2 BHK	1056.0	2.0	39.07	2	3699.810606
1	Chikka Tirupathi	4 Bedroom	2600.0	5.0	120.00	4	4615.384615
2	Uttarahalli	3 BHK	1440.0	2.0	62.00	3	4305.555556
3	Lingadheeranahalli	3 BHK	1521.0	3.0	95.00	3	6245.890861
4	Kothanur	2 BHK	1200.0	2.0	51.00	2	4250.000000
5	Whitefield	2 BHK	1170.0	2.0	38.00	2	3247.863248
6	Old Airport Road	4 BHK	2732.0	4.0	204.00	4	7467.057101
7	Rajaji Nagar	4 BHK	3300.0	4.0	600.00	4	18181.818182
8	Marathahalli	3 BHK	1310.0	3.0	63.25	3	4828.244275
9	other	6 Bedroom	1020.0	6.0	370.00	6	36274.509804

Fig 6.7 : Dimensionality reductions

In fig 6.7, it shows that any location that has less than 10 data points should be tagged as another location. This way the number of categories can be reduced by a huge amount.

```
In [32]: df5[df5.total_sqft/df5.bhk<300].head()
```

Out[32]:

	location	size	total_sqft	bath	price	bhk	price_per_sqft
9	other	6 Bedroom	1020.0	6.0	370.0	6	36274.509804
45	HSR Layout	8 Bedroom	600.0	9.0	200.0	8	33333.333333
58	Murugeshpalya	6 Bedroom	1407.0	4.0	150.0	6	10660.980810
68	Devarachikkanahalli	8 Bedroom	1350.0	7.0	85.0	8	6296.296296
70	other	3 Bedroom	500.0	3.0	100.0	3	20000.000000

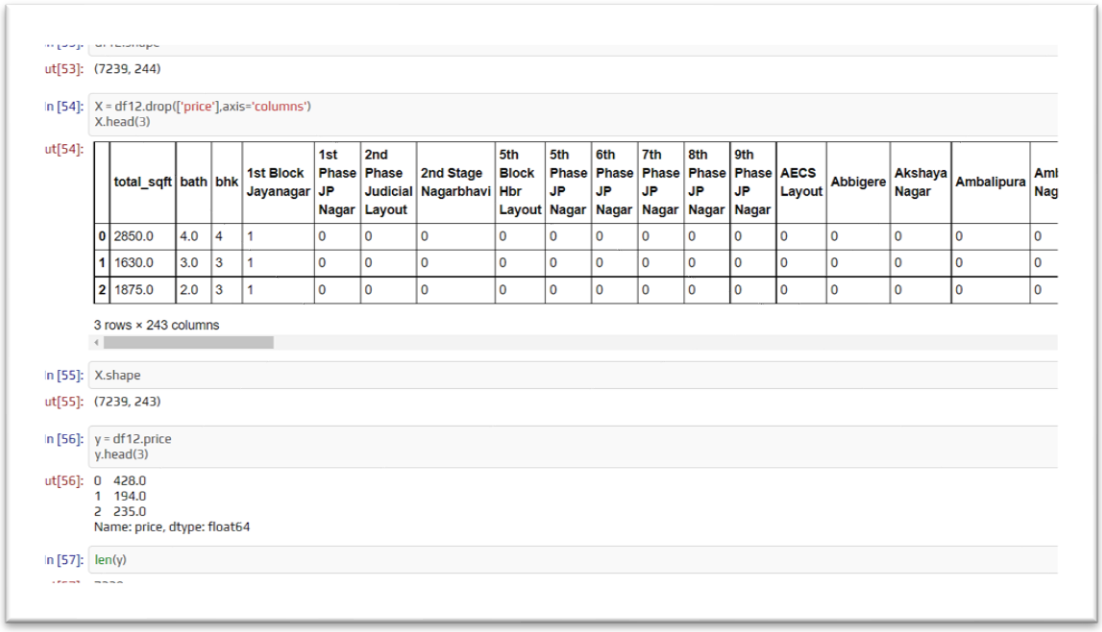
```
In [33]: df5.shape
Out[33]: (13200, 7)

In [34]: df6 = df5[~(df5.total_sqft/df5.bhk<300)]
         df6.shape
Out[34]: (12456, 7)

In [35]: df6.price_per_sqft.describe()
Out[35]: count    12456.000000
         mean      6308.502826
         std       4168.127339
         min        267.829813
         25%       4210.526316
         50%       5294.117647
         75%       6916.666667
         max     176470.588235
         Name: price_per_sqft, dtype: float64
```

Fig 6.8: Outlier removal

In fig 6.8, removing outliers by keeping our minimum threshold per bhk to be 300 sqft. Check the above data points. We have 6 bhk apartments with 1020 sqft. Another one is 8 bhk and the total sqft is 600. These are clear data errors that can be removed safely.



Out[53]: (7239, 244)

In [54]:
```
X = df12.drop(['price'],axis='columns')
X.head(3)
```

Out[54]:

	total_sqft	bath	bhk	1st Block Jayanagar	1st Phase JP Nagar	2nd Phase Judicial Layout	2nd Stage Nagarbhavi	5th Block Hbr Layout	5th Phase JP Nagar	6th Phase JP Nagar	7th Phase JP Nagar	8th Phase JP Nagar	9th Phase JP Nagar	AECS Layout	Abbigere	Akshaya Nagar	Ambalipura	Aml Nag
0	2850.0	4.0	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1630.0	3.0	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1875.0	2.0	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

3 rows × 243 columns

In [55]:
```
X.shape
```

Out[55]: (7239, 243)

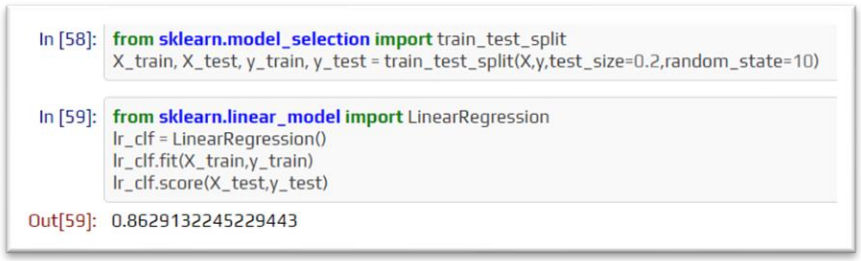
In [56]:
```
y = df12.price
y.head(3)
```

Out[56]:
```
0  428.0
1  194.0
2  235.0
Name: price, dtype: float64
```

In [57]:
```
len(y)
```

Fig 6.9: Build a model

In fig 6.9, model is build for house prediction so that later using grid search the best model is selected.



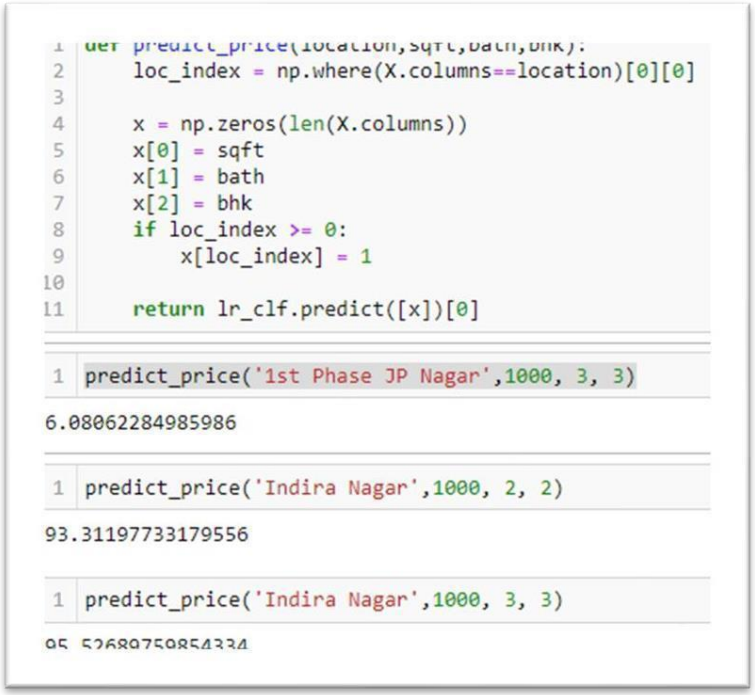
In [58]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.2,random_state=10)
```

In [59]:
```
from sklearn.linear_model import LinearRegression
lr_clf = LinearRegression()
lr_clf.fit(X_train,y_train)
lr_clf.score(X_test,y_test)
```

Out[59]: 0.8629132245229443

Fig 6.10: Accuracy of linear regression model

In fig 6.10, it shows the accuracy of Linear Regression model which is 86%.



```
def predict_price(location,sqft,bath,bhk):
    loc_index = np.where(X.columns==location)[0][0]

    x = np.zeros(len(X.columns))
    x[0] = sqft
    x[1] = bath
    x[2] = bhk
    if loc_index >= 0:
        x[loc_index] = 1

    return lr_clf.predict([x])[0]
```

```
predict_price('1st Phase JP Nagar',1000, 3, 3)
```

```
6.08062284985986
```

```
predict_price('Indira Nagar',1000, 2, 2)
```

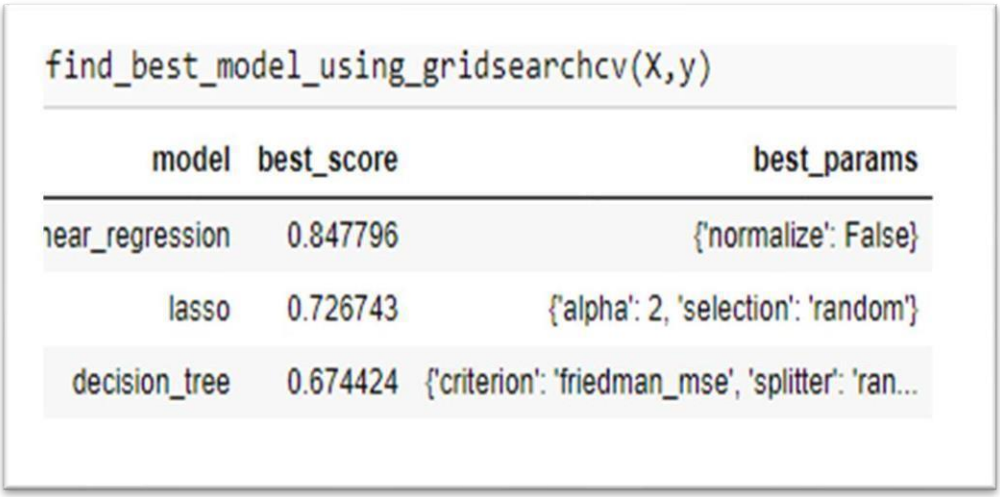
```
93.31197733179556
```

```
predict_price('Indira Nagar',1000, 3, 3)
```

```
95.52689759854334
```

Fig 6.11: Test the model properties

In Fig 6.11, using model properties like sqft, no. of bathrooms and BHK, model is tested topredict the price.



```
find_best_model_using_gridsearchcv(X,y)
```

model	best_score	best_params
linear_regression	0.847796	{'normalize': False}
lasso	0.726743	{'alpha': 2, 'selection': 'random'}
decision_tree	0.674424	{'criterion': 'friedman_mse', 'splitter': 'ran...

Fig 6.12: Best Model

In fig 6.12, Based on the grid search, Linear Regression is the best model as it gives the best score.

CONCLUSION

A machine learning regression project was developed wherein I have learned and obtained several insights about regression models and how they are developed. The various benefits that machine learning provides to the market have made it one of the top contenders in the current market as the field of interest of many organizations worldwide. Most of the organizations are already implementing machine learning technology as it generates more accurate and consistent processes that are less prone to errors. The system uses linear regression model to extract the data and also makes optimal use of machine learning algorithms which satisfies the customer by providing accurate output and preventing the risk of investing in the wrong house.

REFERENCES

[1] Annina S, Mahima SD, Ramesh B, "An Overview of Machine Learning and its Applications", International Journal of Electrical Sciences & Engineering (IJESE). 2015.

[2] Landberg N. The Swedish Housing Market: "An empirical analysis of the price development on the Swedish housing market". Master of Science Thesis. Stockholm: KTH, Engineering and Management; 2015

[3] EliBeracha, Ben T Gilbert, Tyler Kjorstad, Kiplan Womack, "On the Relation between Local Amenties and House Price Dynamics", Journal of real estate Economics, 2016.