PREDICTING HOUSE PRICES USING MACHINE LEARNING

Phase 3 Document

Name: E.Yogapriya **Register Number:**420721104057

INTRODUCTION:

The prediction of house prices is a multifaceted task in the real estate industry, significantly impacting various stakeholders, including homeowners, investors, and policymakers. Understanding the intricate dynamics influencing housing prices is crucial for making informed decisions in this competitive market. By harnessing the power of advanced machine learning techniques, we embark on a journey to develop a sophisticated predictive model capable of accurately estimating house prices based on a comprehensive array of contributing factors. Through the analysis of extensive datasets encompassing property attributes, economic indicators, and market trends, we aim to uncover the underlying patterns and relationships that drive fluctuations in housing prices. The outcomes of this project are expected to provide valuable insights and practical implications for industry professionals and individuals navigating the complex terrain of the real estate sector.

This document outlines the development of a predictive model for house prices using advanced machine learning techniques. By analyzing key factors influencing housing prices, we aim to provide valuable insights for real estate professionals and stakeholders. Through a comprehensive exploration of relevant datasets and the implementation of sophisticated regression methods, our goal is to offer a robust tool for informed decision-making in the ever-evolving real estate landscape.

DATA COLLECTION:

We have collected a comprehensive dataset that encompasses information about various property features, such as location, size, the number of bedrooms, bathrooms, and other relevant attributes. This dataset serves as the foundation for our predictive model.

Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
79545.45857	5.682861322	7.009188143	4.09	23086.8005	1059033.558	208 Michael Ferry Apt. 674
79248.64245	6.002899808	6.730821019	3.09	40173.07217	1505890.915	188 Johnson Views Suite 079
61287.06718	5.86588984	8.51272743	5.13	36882.1594	1058987.988	9127 Elizabeth Stravenue
63345.24005	7.188236095	5.586728665	3.26	34310.24283	1260616.807	USS Barnett
59982.19723	5.040554523	7.839387785	4.23	26354.10947	630943.4893	USNS Raymond
80175.75416	4.988407758	6.104512439	4.04	26748.42842	1068138.074	06039 Jennifer Islands Apt. 443
64698.46343	6.025335907	8.147759585	3.41	60828.24909	1502055.817	4759 Daniel Shoals Suite 442
78394.33928	6.989779748	6.620477995	2.42	36516.35897	1573936.564	972 Joyce Viaduct
59927.66081	5.36212557	6.393120981	2.3	29387.396	798869.5328	USS Gilbert
81885.92718	4.42367179	8.167688003	6.1	40149.96575	1545154.813	Unit 9446 Box 0958
80527.47208	8.093512681	5.0427468	4.1	47224.35984	1707045.722	6368 John Motorway Suite 700
50593.6955	4.496512793	7.467627404	4.49	34343.99189	663732.3969	911 Castillo Park Apt. 717
39033.80924	7.671755373	7.250029317	3.1	39220.36147	1042814.098	209 Natasha Stream Suite 961
73163.66344	6.919534825	5.993187901	2.27	32326.12314	1291331.518	829 Welch Track Apt. 992
69391.38018	5.344776177	8.406417715	4.37	35521.29403	1402818.21	PSC 5330, Box 4420
73091.86675	5.443156467	8.517512711	4.01	23929.52405	1306674.66	2278 Shannon View
79706.96306	5.067889591	8.219771123	3.12	39717.81358	1556786.6	064 Hayley Unions
61929.07702	4.788550242	5.097009554	4.3	24595.9015	528485.2467	5498 Rachel Locks
63508.1943	5.94716514	7.187773835	5.12	35719.65305	1019425.937	Unit 7424 Box 2786
62085.2764	5.739410844	7.091808104	5.49	44922.1067	1030591.429	19696 Benjamin Cape
86294.99909	6.62745694	8.011897853	4.07	47560.77534	2146925.34	030 Larry Park Suite 665
60835.08998	5.551221592	6.517175038	2.1	45574.74166	929247.5995	USNS Brown
64490.65027	4.21032287	5.478087731	4.31	40358.96011	718887.2315	95198 Ortiz Key
60697.35154	6.170484091	7.150536572	6.34	28140.96709	743999.8192	9003 Jay Plains Suite 838
59748.85549	5.339339881	7.748681606	4.23	27809.98654	895737.1334	24282 Paul Valley
56974.47654	8.287562194	7.312879971	4.33	40694.86951	1453974.506	61938 Brady Falls

STEPS FOR LOADING THE DATASET:

Importing Libraries:

Import the required libraries such as Pandas, NumPy, and the machine learning library.

Code:

```
import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from xgboost import XGBRegressor

from sklearn.ensemble import RandomForestRegressor
```

Load the Dataset:

Load the dataset into a Pandas DataFrame. We can typically find house price dataset in CSV format.

Code:

```
housing= pd.read_csv("housing.csv")
housing.head()
```

longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value	ocean_proximity
-122.23	37.88	41	880	129	322	126	8.3252	452600	NEAR BAY
-122.22	37.86	21	7099	1106	2401	1138	8.3014	358500	NEAR BAY
-122.24	37.85	52	1467	190	496	177	7.2574	352100	NEAR BAY
-122.25	37.85	52	1274	235	558	219	5.6431	341300	NEAR BAY
-122.25	37.85	52	1627	280	565	259	3.8462	342200	NEAR BAY
-122.25	37.85	52	919	213	413	193	4.0368	269700	NEAR BAY
-122.25	37.84	52	2535	489	1094	514	3.6591	299200	NEAR BAY
-122.25	37.84	52	3104	687	1157	647	3.12	241400	NEAR BAY

Sample Data

Data Preprocessing:

Data preprocessing is a critical step to ensure the quality and uniformity of the dataset before model training.

Handling Missing Values and Outliers: Identify missing values and outliers in the dataset using methods such as isnull() and describe(), and handle them appropriately based on the specific context of your data.

Converting Categorical Data: Convert categorical data into numerical form using techniques like one-hot encoding or label encoding. This enables the model to process the data effectively.

Feature Scaling: Scale the features if required to bring them to a uniform scale. Common scaling methods include standardization and normalization, which help prevent certain features from dominating the model due to their larger scales.

Code:

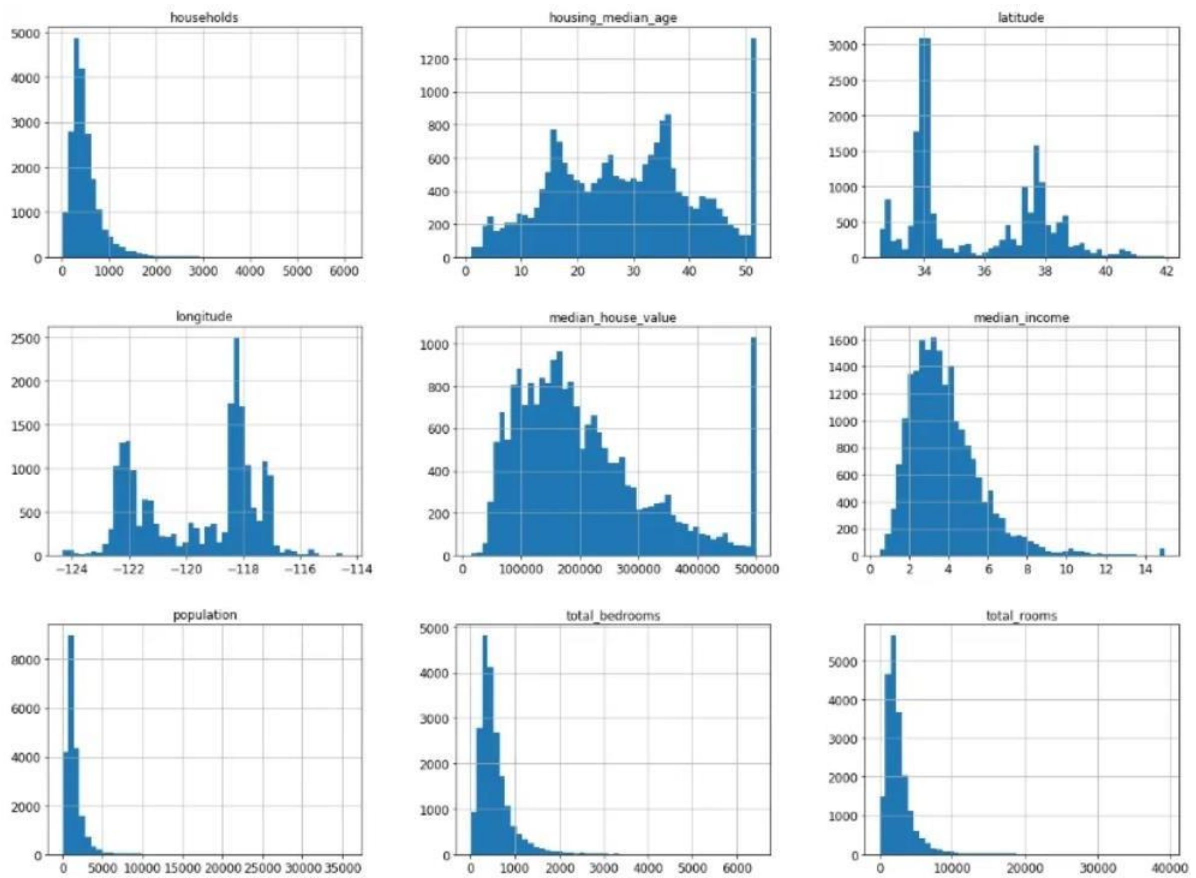
housing.describe()

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households
count	20640.000000	20640.000000	20640.000000	20640.000000	20433.000000	20640.000000	20640.000000
mean	-119.569704	35.631861	28.639486	2635.763081	537.870553	1425.476744	499.539680
std	2.003532	2.135952	12.585558	2181.615252	421.385070	1132.462122	382.329753
min	-124.350000	32.540000	1.000000	2.000000	1.000000	3.000000	1.000000
25%	-121.800000	33.930000	18.000000	1447.750000	296.000000	787.000000	280.000000
50%	-118.490000	34.260000	29.000000	2127.000000	435.000000	1166.000000	409.000000
75%	-118.010000	37.710000	37.000000	3148.000000	647.000000	1725.000000	605.000000
max	-114.310000	41.950000	52.000000	39320.000000	6445.000000	35682.000000	6082.000000

By adopting this approach, we can quickly assess fundamental metrics such as mean, median, and percentiles across multiple features. Utilizing histograms for visualization aids in gaining a comprehensive understanding of the data distribution, facilitating insights into the underlying patterns and characteristics of the dataset.

Code:

```
%matplotlib inline
import matplotlib.pyplot as plt
housing.hist(bins=50, figsize=(20,15))
plt.show()
```



Split the Datasets:

Split the dataset into training and testing sets using functions like train_test_split from the scikit-learn library.We will divide the datasets into train and test split with 80% of the data for model building and 20% for testing the model.

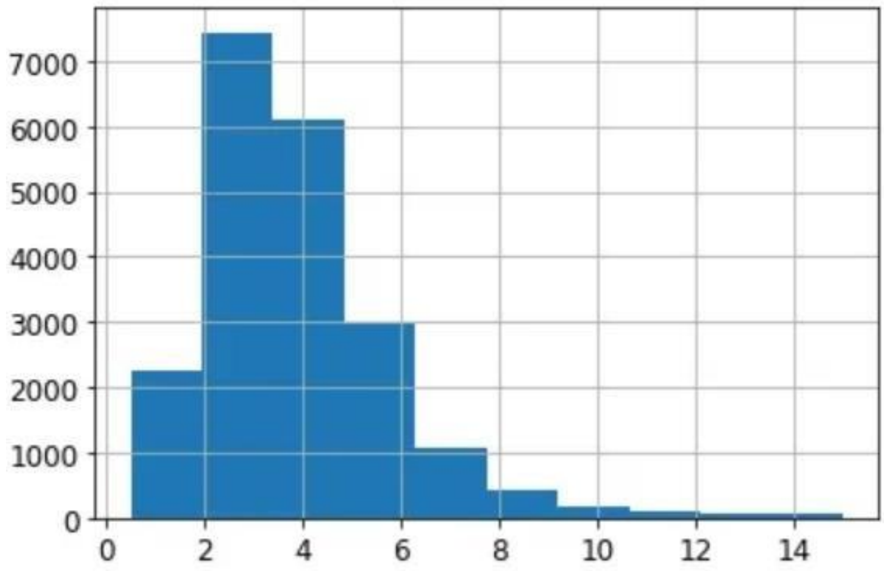
Code:

```
from sklearn.model_selection import train_test_split

train_set,test_set=train_test_split(housing,test_size=0.2,random_state
=42)
```

In this case, we employed random sampling to generate training and testing datasets. Typically, the median income of a neighbourhood serves as a robust indicator of the wealth distribution in the area. Therefore, we aim to ensure that the test dataset accurately represents the diverse income categories, requiring us to convert it into categorical variables and implement stratified sampling instead of random sampling.

```
housing["median_income"].hist()
```

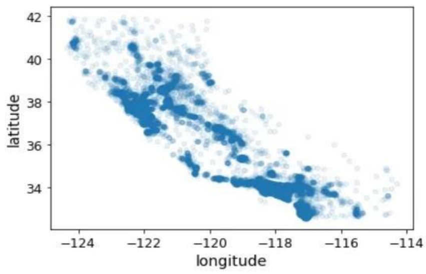


Exploring the Data:

Data exploration involves understanding the dataset's structure, summarizing key statistics, visualizing data distributions, analyzing variable relationships, and assessing feature importance. This process facilitates the identification of patterns and insights crucial for informed decision-making and accurate modeling.

Code:

```
housing.plot(kind="scatter", x="longitude", y="latitude", alpha=0.1)
```



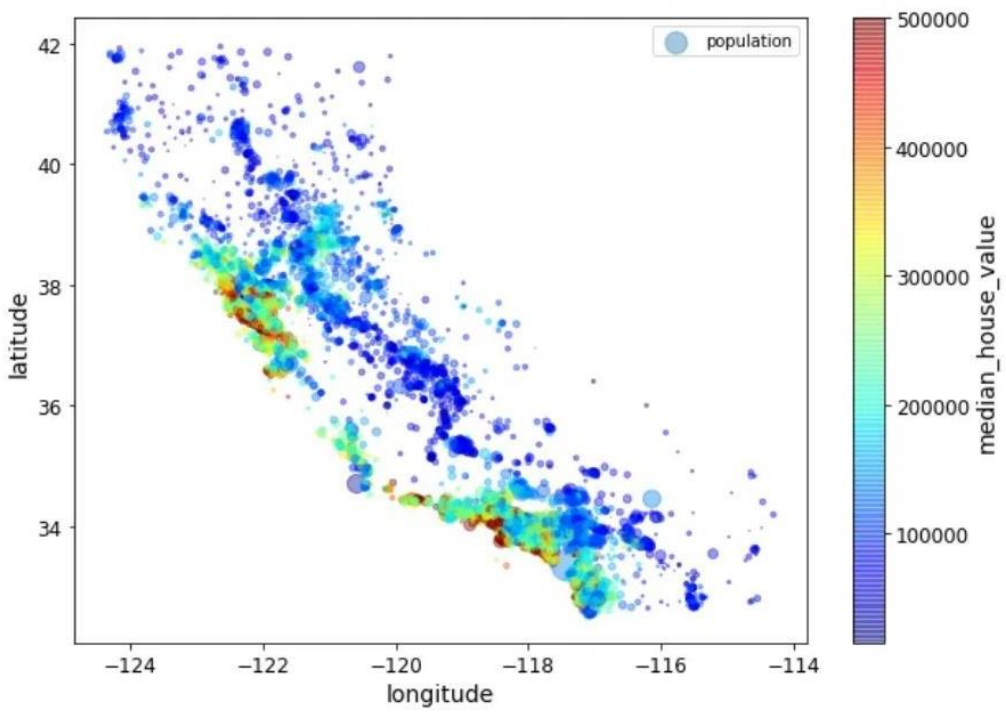
Code:

```
import matplotlib.pyplot as plt

housing.plot(kind="scatter", x="longitude", y="latitude", alpha=0.4,
s=housing["population"]/100, label="population", figsize=(10,7),
c="median_house_value",  cmap=plt.get_cmap("jet"),  colorbar=True,
sharex=False)

plt.legend()

plt.show()
```



Housing Prices

CONCLUSION:

In our pursuit of constructing a house price prediction model, we have embarked on a crucial journey, commencing with the loading and preprocessing of the dataset. We have navigated through fundamental steps, commencing with the importation of essential libraries to facilitate data manipulation and analysis. Understanding the data's structure, attributes, and potential issues through exploratory data analysis (EDA) is imperative for making informed decisions. Data preprocessing has emerged as a pivotal component of this process, encompassing the cleansing, transformation, and refinement of the dataset to meet the requisites of machine learning algorithms. With these fundamental steps, we establish a solid foundation.