Predicting House Prices using Machine Learning

# INTRODUCTION

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.In the case of house price prediction, we can use historical data on various features of a house, such as its location, size, and amenities, to train a machine-learning model. Once the model is trained, it can analyze new data on a given house and make a prediction of its market value

In this phase loading and preprocessing of house price prediction is going to be done.

# DATASET

The data is obtained from <https://www.kaggle.com/datasets/vedavyasv/usa-housing>

# COLUMNS USED

##### 

##### From USA\_Housing.csv data the following columns are used

* **Avg. Area Income**
* **Avg. Area House Age**
* **Avg. Area Number of Rooms**
* **Avg. Area Number of Bedrooms**
* **Area Population**
* **Price**
* **Address**

# LIBRARIES USED

The Python 3 environment comes with many helpful analytics libraries installed and several helpful packages to load.

The essential libraries used in this project are :

* Importing OS (for kaggle inputs)
* Numpy and Pandas libraries
* Matplotlib
* seaborn

### **Loading Dataset**

dataset = pd.read\_csv('/kaggle/input/usa-housing/USA\_Housing.csv')

**dataset.info()**

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 5000 entries, 0 to 4999

Data columns (total 7 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Avg. Area Income 5000 non-null float64

1 Avg. Area House Age 5000 non-null float64

2 Avg. Area Number of Rooms 5000 non-null float64

3 Avg. Area Number of Bedrooms 5000 non-null float64

4 Area Population 5000 non-null float64

5 Price 5000 non-null float64

6 Address 5000 non-null object

dtypes: float64(6), object(1)

memory usage: 273.6+ KB

**dataset.columns**

Index(['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',

'Avg. Area Number of Bedrooms', 'Area Population', 'Price', 'Address'],

dtype='object')

|  | Avg. Area Income | Avg. Area House Age | Avg. Area Number of Rooms | Avg. Area Number of Bedrooms | Area Population | Price |
| --- | --- | --- | --- | --- | --- | --- |
| count | 5000.000000 | 5000.000000 | 5000.000000 | 5000.000000 | 5000.000000 | 5.000000e+03 |
| mean | 68583.108984 | 5.977222 | 6.987792 | 3.981330 | 36163.516039 | 1.232073e+06 |
| std | 10657.991214 | 0.991456 | 1.005833 | 1.234137 | 9925.650114 | 3.531176e+05 |
| min | 17796.631190 | 2.644304 | 3.236194 | 2.000000 | 172.610686 | 1.593866e+04 |
| 25% | 61480.562388 | 5.322283 | 6.299250 | 3.140000 | 29403.928702 | 9.975771e+05 |
| 50% | 68804.286404 | 5.970429 | 7.002902 | 4.050000 | 36199.406689 | 1.232669e+06 |
| 75% | 75783.338666 | 6.650808 | 7.665871 | 4.490000 | 42861.290769 | 1.471210e+06 |
| max | 107701.748378 | 9.519088 | 10.759588 | 6.500000 | 69621.713378 | 2.469066e+06 |

**dataset.describe()**

# **Pre-Processing of Data**

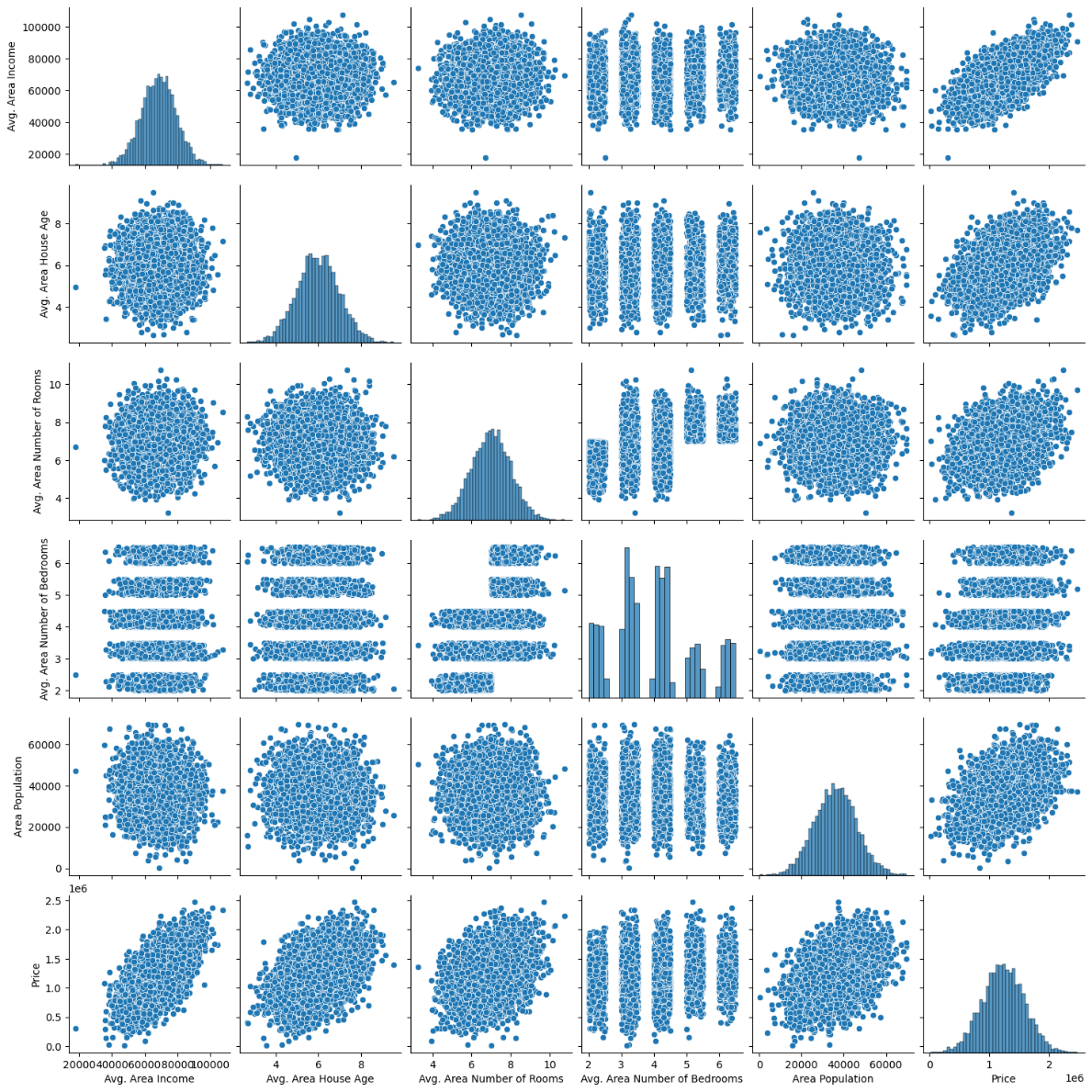
* now we are going create joint plot for house prices prediction
* lets take x-axis as Avg. Area House Age
* y-axis as Price
* sns.jointplot(dataset, x='Avg. Area House Age', y='Price', kind='hex')
* <seaborn.axisgrid.JointGrid at 0x7dbe246100a0>



* now we are going create joint plot for house prices prediction
* lets take x-axis as Avg. Area Income
* y-axis as Price
* sns.jointplot(dataset, x='Avg. Area Income', y='Price')
* <seaborn.axisgrid.JointGrid at 0x7dbe1333c250>



* Now we going to create pairplot for house prices prediction
* All columns figures are given below
* plt.figure(figsize=(12,8))
* sns.pairplot(dataset)
* <seaborn.axisgrid.PairGrid at 0x7dbe1333c340>
* <Figure size 1200x800 with 0 Axes>



dataset.hist(figsize=(10,8))

array([[<Axes: title={'center': 'Avg. Area Income'}>,

<Axes: title={'center': 'Avg. Area House Age'}>],

[<Axes: title={'center': 'Avg. Area Number of Rooms'}>,

<Axes: title={'center': 'Avg. Area Number of Bedrooms'}>],

[<Axes: title={'center': 'Area Population'}>,

<Axes: title={'center': 'Price'}>]], dtype=object)

