# House_price_prediction

January 6, 2023

[1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

%matplotlib inline
```

[2]:

```
HouseDF = pd.read_csv('USA_Housing.csv')
HouseDF.head()
```

[2]:

```
   Avg. Area Income  Avg. Area House Age  Avg. Area Number of Rooms  \
0      79545.458574             5.682861                   7.009188
1      79248.642455             6.002900                   6.730821
2      61287.067179             5.865890                   8.512727
3      63345.240046             7.188236                   5.586729
4      59982.197226             5.040555                   7.839388

   Avg. Area Number of Bedrooms  Area Population         Price  \
0                          4.09     23086.800503  1.059034e+06
1                          3.09     40173.072174  1.505891e+06
2                          5.13     36882.159400  1.058988e+06
3                          3.26     34310.242831  1.260617e+06
4                          4.23     26354.109472  6.309435e+05

                                             Address
0  208 Michael Ferry Apt. 674\nLaurabury, NE 3701…
1  188 Johnson Views Suite 079\nLake Kathleen, CA…
2  9127 Elizabeth Stravenue\nDanieltown, WI 06482…
3                          USS Barnett\nFPO AP 44820
4                         USNS Raymond\nFPO AE 09386
```

[3]:

```
HouseDF.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
```

[4]:

```
HouseDF.describe()
```

[4]:

```
       Avg. Area Income  Avg. Area House Age  Avg. Area Number of Rooms  \
count       5000.000000          5000.000000                5000.000000
mean       68583.108984             5.977222                   6.987792
std        10657.991214             0.991456                   1.005833
min        17796.631190             2.644304                   3.236194
25%        61480.562388             5.322283                   6.299250
50%        68804.286404             5.970429                   7.002902
75%        75783.338666             6.650808                   7.665871
max       107701.748378             9.519088                  10.759588

       Avg. Area Number of Bedrooms  Area Population         Price
count                   5000.000000      5000.000000  5.000000e+03
mean                       3.981330     36163.516039  1.232073e+06
std                        1.234137      9925.650114  3.531176e+05
min                        2.000000       172.610686  1.593866e+04
25%                        3.140000     29403.928702  9.975771e+05
50%                        4.050000     36199.406689  1.232669e+06
75%                        4.490000     42861.290769  1.471210e+06
max                        6.500000     69621.713378  2.469066e+06
```

[5]:

```
HouseDF.columns
```

[5]:

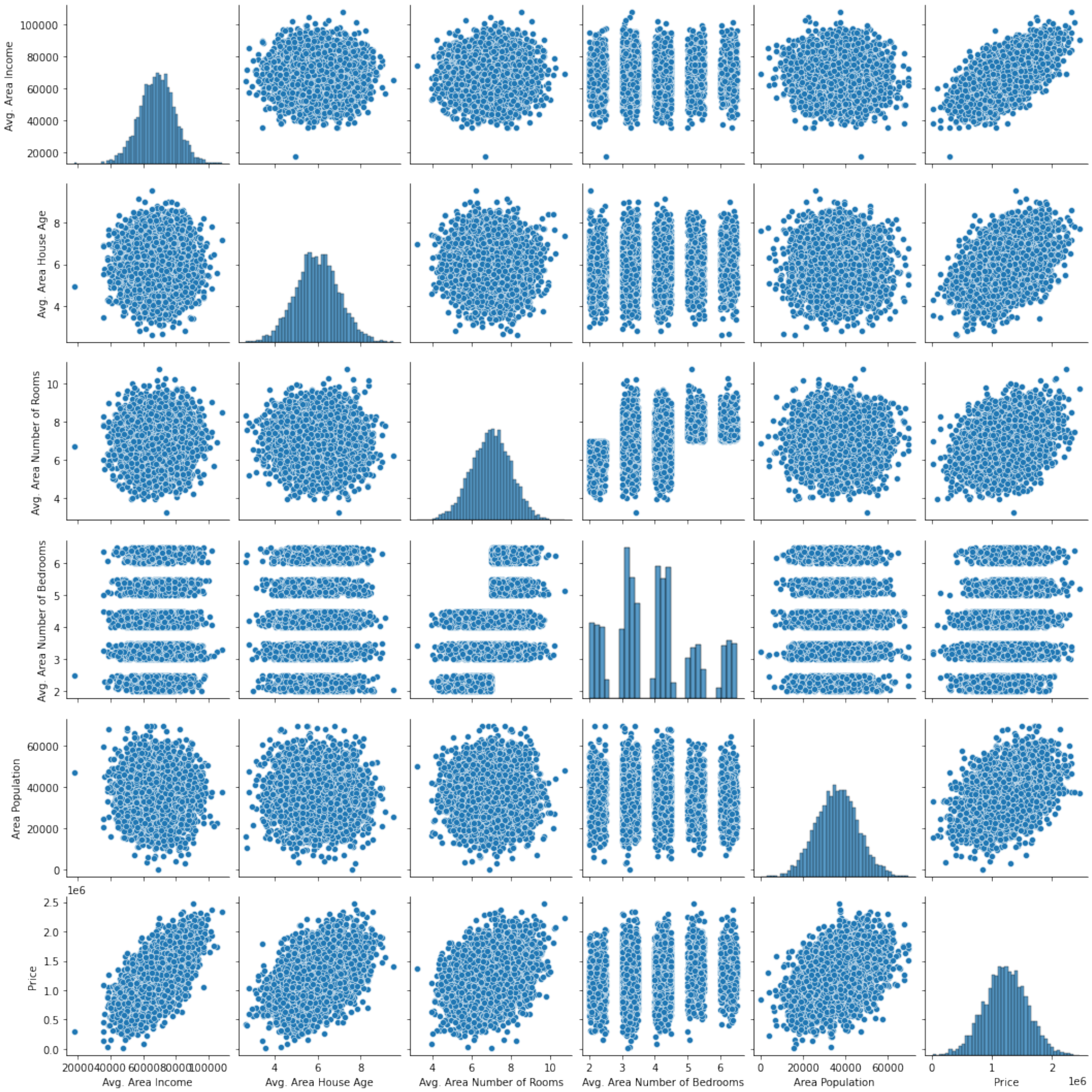
```
Index(['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
       'Avg. Area Number of Bedrooms', 'Area Population', 'Price', 'Address'],
      dtype='object')
```

[6]:

```
sns.pairplot(HouseDF)
```

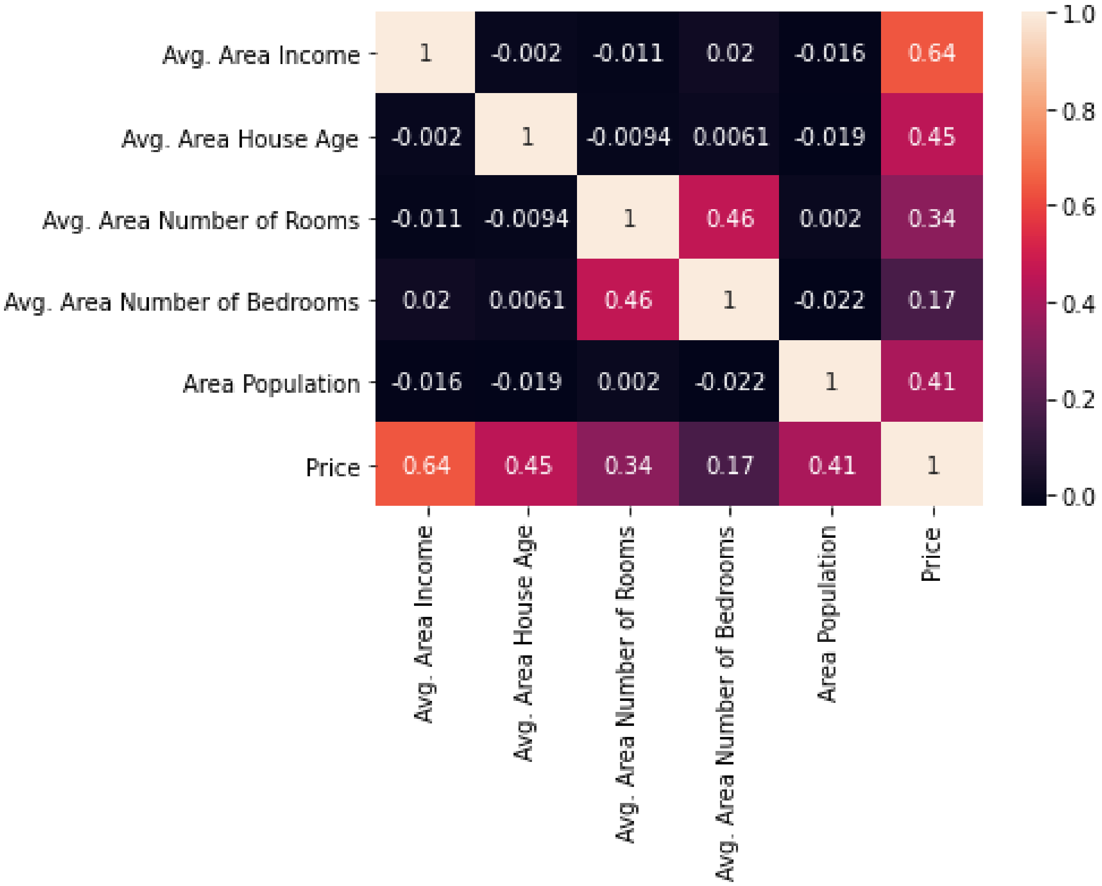
[6]:

```
<seaborn.axisgrid.PairGrid at 0x277a7b15c70>
```



```
[7]: sns.heatmap(HouseDF.corr(), annot=True)
```

```
[7]: <AxesSubplot:>
```



[8]:
```
X = HouseDF[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of␣
 ↪Rooms',
               'Avg. Area Number of Bedrooms', 'Area Population']]

Y = HouseDF['Price']
```

[9]:
```
from sklearn.model_selection import train_test_split

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.4,␣
 ↪random_state=101)
X_train
```

[9]:
```
      Avg. Area Income  Avg. Area House Age  Avg. Area Number of Rooms  \
1303      68091.179676             5.364208                   7.502956
1051      75729.765546             5.580599                   7.642973
4904      70885.420819             6.358747                   7.250241
931       73386.407340             4.966360                   7.915453
4976      75046.313791             5.351169                   7.797825
…                  …                    …                          …
```

```
4171      56610.642563           4.846832                   7.558137
599       70596.850945           6.548274                   6.539986
1361      55621.899104           3.735942                   6.868291
1547      63044.460096           5.935261                   5.913454
4959      75078.791516           7.644779                   8.440726

      Avg. Area Number of Bedrooms  Area Population
1303                          3.10     44557.379656
1051                          4.21     29996.018448
4904                          5.42     38627.301473
931                           4.30     38413.490484
4976                          5.23     34107.888619
…                              …              …
4171                          3.29     25494.740298
599                           3.10     51614.830136
1361                          2.30     63184.613147
1547                          4.10     32725.279544
4959                          4.33     56148.449322

[3000 rows x 5 columns]
```

```
[10]: from sklearn.linear_model import LinearRegression
```

```
[11]: lm = LinearRegression()
      lm.fit(X_train,Y_train)
```

```
[11]: LinearRegression()
```

```
[12]: print(lm.intercept_)
```

```
-2640159.7968519107
```

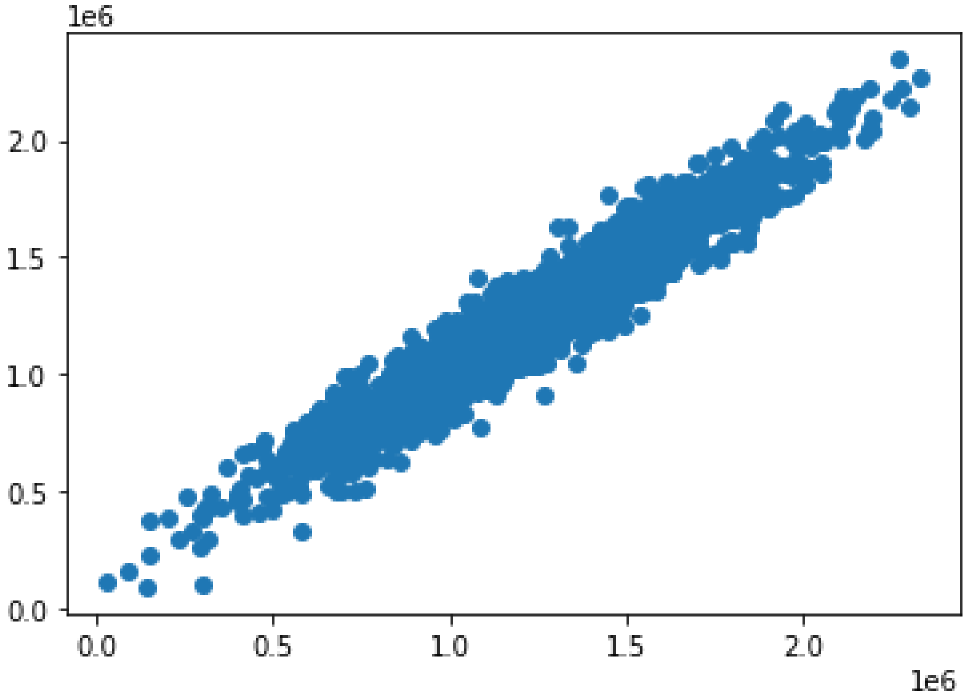
```
[13]: coeff_df = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient'])
      coeff_df
```

```
[13]:                                Coefficient
      Avg. Area Income                 21.528276
      Avg. Area House Age          164883.282027
      Avg. Area Number of Rooms    122368.678027
      Avg. Area Number of Bedrooms   2233.801864
      Area Population                  15.150420
```

```
[14]: predictions = lm.predict(X_test)
```

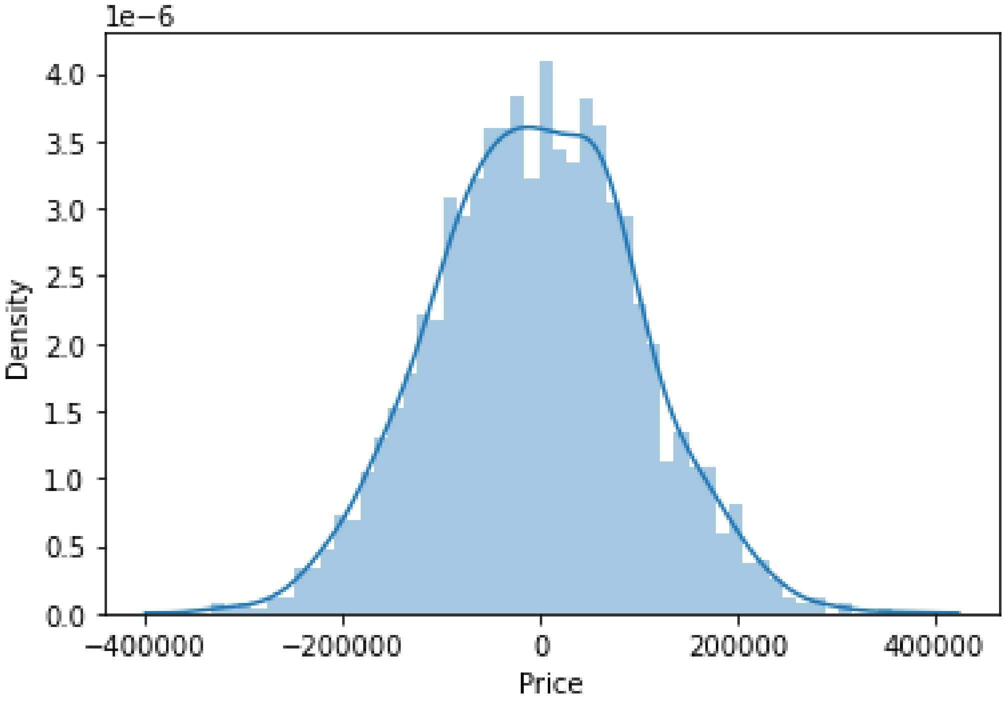
```
[15]: plt.scatter(Y_test,predictions)
```

```
[15]: <matplotlib.collections.PathCollection at 0x277ac39ad00>
```



[16]:
```
sns.distplot((Y_test-predictions),bins=50);
```

```
D:\A\New folder\lib\site-packages\seaborn\distributions.py:2557: FutureWarning:
`distplot` is a deprecated function and will be removed in a future version.
Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



```
[17]: from sklearn import metrics

      print('MAE:', metrics.mean_absolute_error(Y_test, predictions))
      print('MSE:', metrics.mean_squared_error(Y_test, predictions))
      print('RMSE:', np.sqrt(metrics.mean_squared_error(Y_test, predictions)))
```

```
MAE: 82288.22251914955
MSE: 10460958907.209503
RMSE: 102278.82922291153
```

```
[ ]:
```