

SmartInternz (Evening Batch)

Assignment-2

Name: Prakhar Agarwal

Reg No.: 21BIT0034

Email ID: prakhar.agarwal2021@vitstudent.ac.in

1. Download the dataset

```
from google.colab import files
uploaded = files.upload()

<IPython.core.display.HTML object>

Saving House Price India.csv.zip to House Price India.csv.zip
```

2. Load the dataset.

```
import zipfile

with zipfile.ZipFile('House Price India.csv.zip', 'r') as zip_ref:
    zip_ref.extractall('data_folder')

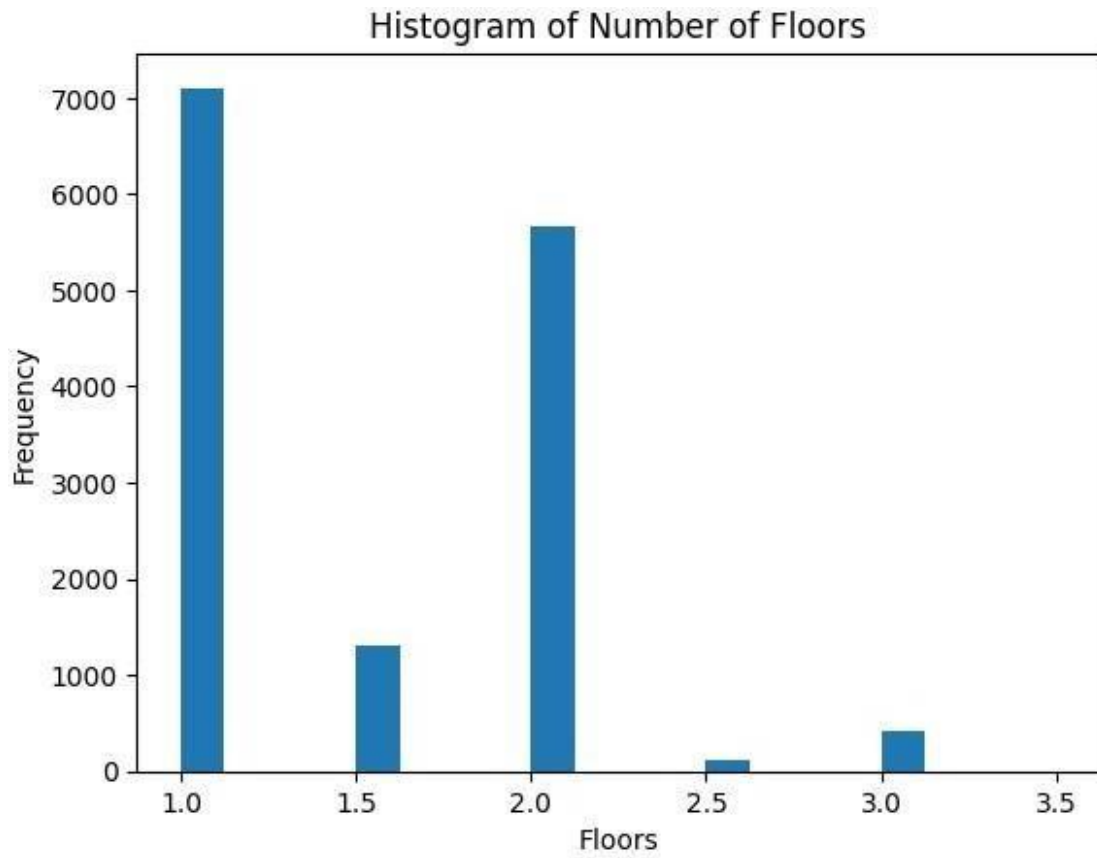
import pandas as pd

# Load the CSV file into a DataFrame
df = pd.read_csv('data_folder/House Price India.csv')
```

3. Perform the Below Visualizations. • Univariate Analysis

```
import matplotlib.pyplot as plt

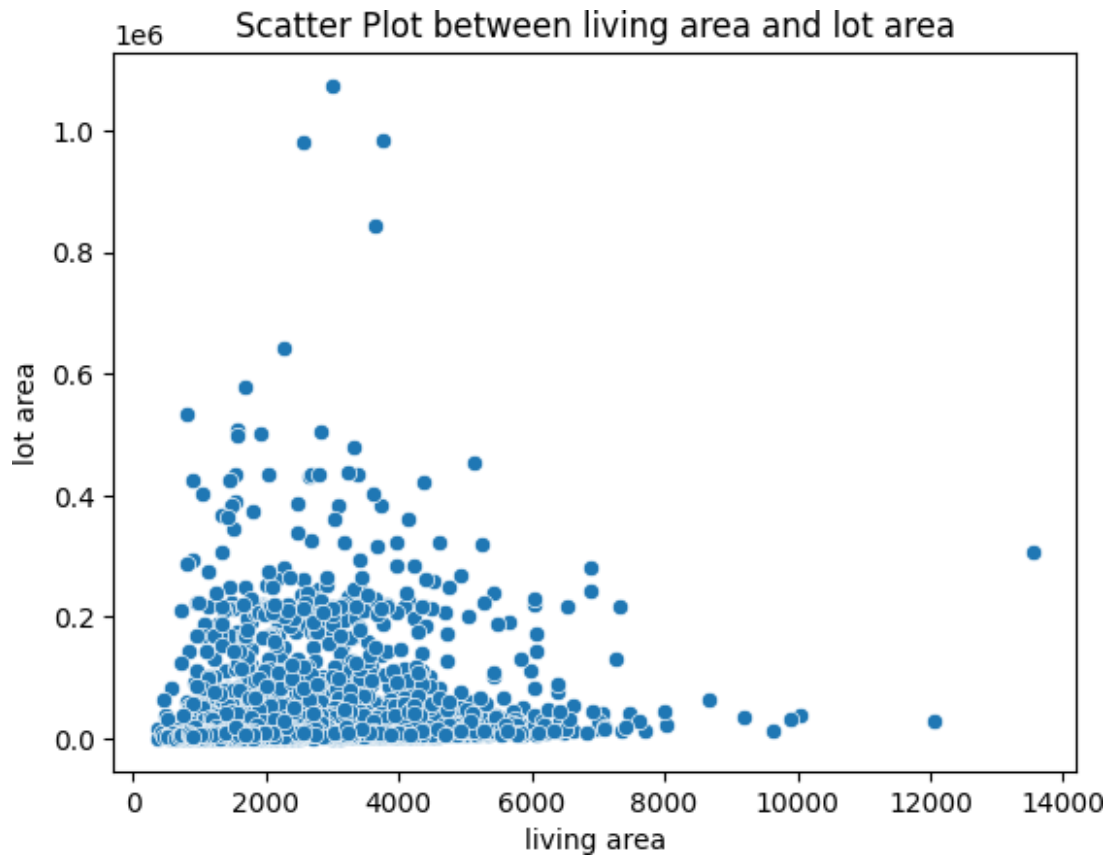
plt.hist(df['number of floors'], bins=20)
plt.xlabel('Floors')
plt.ylabel('Frequency')
plt.title('Histogram of Number of Floors')
plt.show()
```



- Bi - Variate Analysis

```
import seaborn as sns
```

```
sns.scatterplot(x='living area', y='lot area', data=df)  
plt.xlabel('living area')  
plt.ylabel('lot area')  
plt.title('Scatter Plot between living area and lot area')  
plt.show()
```



• Multivariate Analysis

```
descriptive_stats = df.describe()
print(descriptive_stats)
```

	id	Date	number of bedrooms	number of bathrooms \
count	1.462000e+04	14620.000000	14620.000000	14620.000000
mean	6.762821e+09	42604.538646	3.379343	2.129583
std	6.237575e+03	67.347991	0.938719	0.769934
min	6.762810e+09	42491.000000	1.000000	0.500000
25%	6.762815e+09	42546.000000	3.000000	1.750000
50%	6.762821e+09	42600.000000	3.000000	2.250000
75%	6.762826e+09	42662.000000	4.000000	2.500000
max	6.762832e+09	42734.000000	33.000000	8.000000

	living area	lot area	number of floors	waterfront
present \				
count	14620.000000	1.462000e+04	14620.000000	
14620.000000				
mean	2098.262996	1.509328e+04	1.502360	
0.007661				
std	928.275721	3.791962e+04	0.540239	
0.087193				
min	370.000000	5.200000e+02	1.000000	
0.000000				
25%	1440.000000	5.010750e+03	1.000000	
0.000000				
50%	1930.000000	7.620000e+03	1.500000	
0.000000				
75%	2570.000000	1.080000e+04	2.000000	
0.000000				
max	13540.000000	1.074218e+06	3.500000	
1.000000				

	number of views	condition of the house	...	Built Year	\
count	14620.000000	14620.000000	...	14620.000000	
mean	0.233105	3.430506	...	1970.926402	
std	0.766259	0.664151	...	29.493625	
min	0.000000	1.000000	...	1900.000000	
25%	0.000000	3.000000	...	1951.000000	
50%	0.000000	3.000000	...	1975.000000	
75%	0.000000	4.000000	...	1997.000000	
max	4.000000	5.000000	...	2015.000000	

	Renovation Year	Postal Code	Lattitude	Longitude	\
count	14620.000000	14620.000000	14620.000000	14620.000000	
mean	90.924008	122033.062244	52.792848	-114.404007	
std	416.216661	19.082418	0.137522	0.141326	
min	0.000000	122003.000000	52.385900	-114.709000	
25%	0.000000	122017.000000	52.707600	-114.519000	
50%	0.000000	122032.000000	52.806400	-114.421000	
75%	0.000000	122048.000000	52.908900	-114.315000	
max	2015.000000	122072.000000	53.007600	-113.505000	

	living_area_renov	lot_area_renov	Number of schools nearby	\
count	14620.000000	14620.000000	14620.000000	
mean	1996.702257	12753.500068	2.012244	
std	691.093366	26058.414467	0.817284	
min	460.000000	651.000000	1.000000	
25%	1490.000000	5097.750000	1.000000	
50%	1850.000000	7620.000000	2.000000	
75%	2380.000000	10125.000000	3.000000	
max	6110.000000	560617.000000	3.000000	

	Distance from the airport	Price
count	14620.000000	1.462000e+04
mean	64.950958	5.389322e+05
std	8.936008	3.675324e+05
min	50.000000	7.800000e+04
25%	57.000000	3.200000e+05
50%	65.000000	4.500000e+05
75%	73.000000	6.450000e+05
max	80.000000	7.700000e+06

[8 rows x 23 columns]

5. Handle the Missing values

```
df.fillna(df.mean(), inplace=True)
```