HOUSING

DATA ANALYSIS

Prepared by :

chandana T M

DA/DS ONLINE

batch 28

INTRODUCTION

The objective of this project is to perform exploratory data analysis (EDA) on a housing dataset to uncover insights and patterns that can inform decision-making. The dataset includes various attributes of housing units such as price, location, size, and features. By conducting EDA, we aim to understand the underlying structure of the data, identify key factors affecting housing prices, and provide valuable insights for stakeholders such as real estate professionals and potential buyers.

AIM

The aim of this project is to utilize exploratory data analysis techniques to:

- Understand the key characteristics and distributions of the housing dataset.
- Identify and address data quality issues.
- Derive meaningful metrics and insights that can help in predicting housing prices.
- Highlight significant trends, correlations, and patterns in the data to guide strategic decision-making.

Business Problem / Problem Statement

In the competitive real estate market, accurately determining the factors that influence housing prices is crucial for both buyers and sellers. The problem is to identify which features of the housing data—such as location, size, and amenities—most significantly impact the price of a property. Understanding these factors will assist stakeholders in making informed decisions, optimizing property value assessments, and formulating pricing strategies. The dataset provides a comprehensive view of housing units, but extracting actionable insights requires careful analysis to uncover hidden trends and relationships.

Project Workflow

The project workflow involves several key steps:

- Data Collection: Gather the housing dataset from reliable sources.
- Data Understanding: Explore and understand the structure and contents of the data.
- Data Cleaning: Address missing values, outliers, and inconsistent data.
- Bivariate Analysis: Investigate relationships between pairs of variables.
- Multivariate Analysis: Explore interactions between multiple variables.
- Insights: Synthesize findings to provide actionable insights and recommendations.

Data Understanding

The housing dataset comprises various attributes, including but not limited to, property prices, location details, size (in square feet), number of bedrooms and bathrooms, and additional features such as garden or garage. Initial exploration involves examining the dataset's summary statistics, distribution of key variables, and basic visualizations. Understanding the data's structure helps in identifying the types of variables (categorical, numerical) and their potential relationships. This step also includes checking for the completeness and integrity of the data.

Data Cleaning - Missing Values Imputation, Outliers, Handling Inconsistent Values

Data cleaning involves several processes:

Missing Values Imputation: Identify missing values and use appropriate methods to impute them, such as mean, median, or mode for numerical values, and mode or a specific category for categorical values.

Outliers: Detect outliers using statistical methods (e.g., Z-score, IQR) and decide whether to remove them or transform them, depending on their impact on the analysis.

Handling Inconsistent Values: Correct inconsistencies in categorical values (e.g., different spellings or formats) and standardize units and formats across the dataset. This ensures uniformity and accuracy in the analysis.

lentify and address missing values in the dataset through imputation or removal, ensuring data completeness.

etect and rectify any inconsistencies or anomalies in the data, such as erroneous entries or irregular formatting.

```
d1=df['sqft_living'].sort_values(ascending=True)
```

```
d1
```

```
3778    370.0
2919    380.0
2416    420.0
1219    430.0
4184    490.0
        ...
59        NaN
60        NaN
61        NaN
62        NaN
63        NaN
Name: sqft_living, Length: 4600, dtype: float64
```

```
d2=df.sqft_living.median()
```

```
d2
```

```
1980.0
```

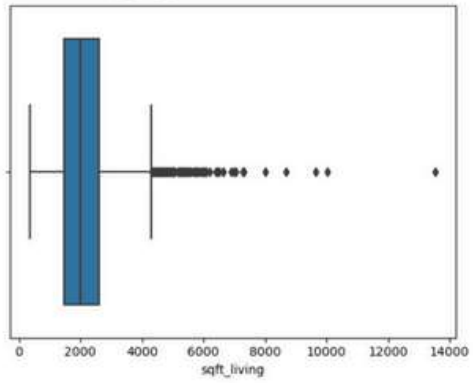
```
df.sqft_living.fillna(d2,inplace=True)
```

```
df.isnull().sum()
```

```
date           0
price          0
bedrooms       0
bathrooms      0
sqft_living    0
```

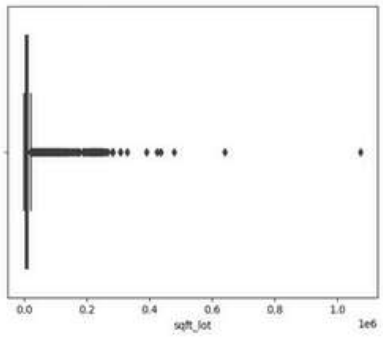
```
sns.boxplot(xsdf['sqft_living'])
```

```
<Axes: xlabel='sqft_living'>
```




```
sns.boxplot(xsdf['sqft_lot'])
```

```
<Axes: xlabel='sqft_lot'>
```




```
=df['sqft_lot'].sort_values(ascending=True)
```

```
608    638.0
15     681.0
913    704.0
004    746.0
247    747.0
       ...
       NaN
0      NaN
1      NaN
2      NaN
3      NaN
ame: sqft_lot, Length: 4600, dtype: float64
```

```
1sdf.sqft_lot.median()
```

```
1
```

```
683.5
```

```
df.sqft_lot.fillna(c1,inplace=True)
```

```
df.isnull().sum()
```

```
date           0
price          0
bedrooms       0
bathrooms      0
sqft_living    0
sqft_lot       0
floors         0
waterfront     0
```

```
1931., 1901., 1907., 1915., 1935., 1933., 1934.,   nan])
```

```
c2sdf.yr_built.mean()
```

```
c2
```

```
1970.8088267424077
```

```
df.yr_built.fillna(c2,inplace=True)
```

```
df.isnull().sum()
```

```
date           0
price          0
bedrooms       0
bathrooms      0
sqft_living    0
sqft_lot       0
floors         0
waterfront     0
view           0
condition      0
sqft_above     0
sqft_basement  0
yr_built       0
yr_renovated   0
street         0
city          57
statezip       0
```

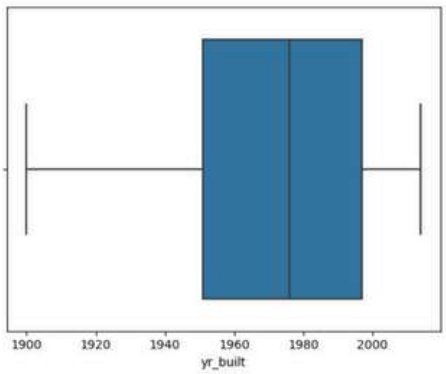
```
'SeaTac', 'Medina', 'Enumclaw', 'Snoqualmie Pass', 'Pacific',
'Beaux Arts Village', 'Preston', 'Milton'], dtype=object)
```

```
df.isnull().sum()
```

```
date           0
price          0
bedrooms       0
bathrooms      0
sqft_living    0
sqft_lot       0
floors         0
waterfront     0
view           0
```

```
sns.boxplot(xsdf['yr_built'])
```

```
<Axes: xlabel='yr_built'>
```

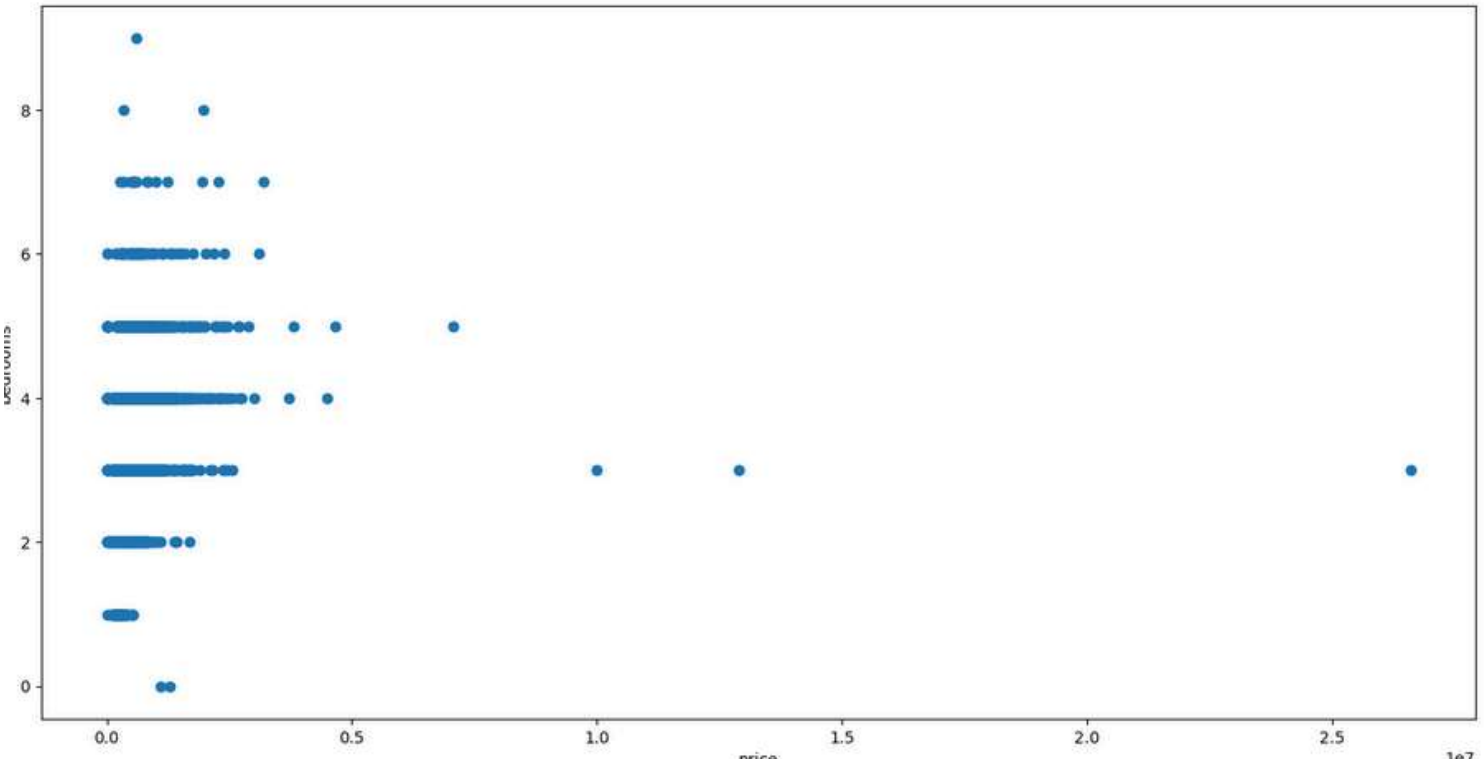


Bivariate Analysis

Bivariate analysis explores relationships between two variables. Techniques include:

Scatter Plots: To visualize relationships between numerical variables like price and size.
Correlation Analysis: Calculating correlation coefficients to assess the strength and direction of relationships.
Box Plots: To compare distributions of numerical variables across different categories.
Bivariate analysis helps in identifying trends, correlations, and potential causations between variables.

```
fig, ax = plt.subplots(figsize=(16,8))
ax.scatter(df['price'], df['bedrooms'])
ax.set_xlabel('price')
ax.set_ylabel('bedrooms')
plt.show()
```



Multivariate Analysis

correlation

Correlation is used to find the relationship between two variables which is important in real life because we can predict the value of one variable with the help of other variables, who is being correlated with it. It is a type of Bivariate statistics since two variables are involved here.

Correlation is a statistical measure that expresses the extent to which two variables are linearly related.

```
q1=data.quantile(0.25)
```

```
q1
```

```
price          322500.00
bedrooms            3.00
bathrooms           1.75
sqft_living      1470.00
sqft_lot         5001.00
floors              1.00
waterfront          0.00
view                0.00
condition           3.00
sqft_above       1190.00
sqft_basement       0.00
yr_built         1951.00
yr_renovated        0.00
Name: 0.25, dtype: float64
```

```
q3=data.quantile(0.75)
```

```
q3
```

```
price          655000.0
bedrooms             4.0
bathrooms            2.5
sqft_living       2610.0
sqft_lot         11000.0
floors               2.0
waterfront           0.0
view                 0.0
condition            4.0
sqft_above        2300.0
```

```
IQR=q3-q1
print(IQR)
```

```
price          332500.00
bedrooms            1.00
bathrooms           0.75
sqft_living      1140.00
sqft_lot         5999.00
floors              1.00
waterfront          0.00
view                0.00
condition           1.00
sqft_above       1110.00
sqft_basement     610.00
yr_built           46.00
yr_renovated     1999.00
dtype: float64
```

```
print((data < (q1 - 1.5 * IQR)) | (data > (q3 + 1.5 * IQR)))
```

```
      price  bedrooms  bathrooms  sqft_living  sqft_lot  floors  waterfront  \
0     False     False      False        False     False   False       False
1      True     False      False        False     False   False       False
2     False     False      False        False     False   False       False
3     False     False      False        False     False   False       False
4     False     False      False        False     False   False       False
...     ...       ...        ...          ...       ...     ...         ...
4595  False     False      False        False     False   False       False
4596  False     False      False        False     False   False       False
4597  False     False      False        False     False   False       False
4598  False     False      False        False     False   False       False
4599  False     False      False        False     False   False       False

      view  condition  sqft_above  sqft_basement  yr_built  yr_renovated
0    False      False       False          False     False         False
1     True      False       False          False     False         False
...    ...        ...         ...            ...       ...           ...
```

```
vv=(data < (q1 - 1.5 * IQR)) | (data > (q3 + 1.5 * IQR))
```

```
v.sum()
```

```
price          241
bedrooms       318
bathrooms      161
sqft_living    134
sqft_lot       565
floors           0
waterfront      33
view           460
condition        6
sqft_above     121
sqft_basement   82
yr_built         0
yr_renovated     0
dtype: int64
```

```
df.shape
```

```
(4600, 18)
```

```
data_clean = data[~((data < (q1 - 1.5 * IQR)) | (data > (q3 + 1.5 * IQR))).any(axis=1)]
```

```
data_clean.shape
```

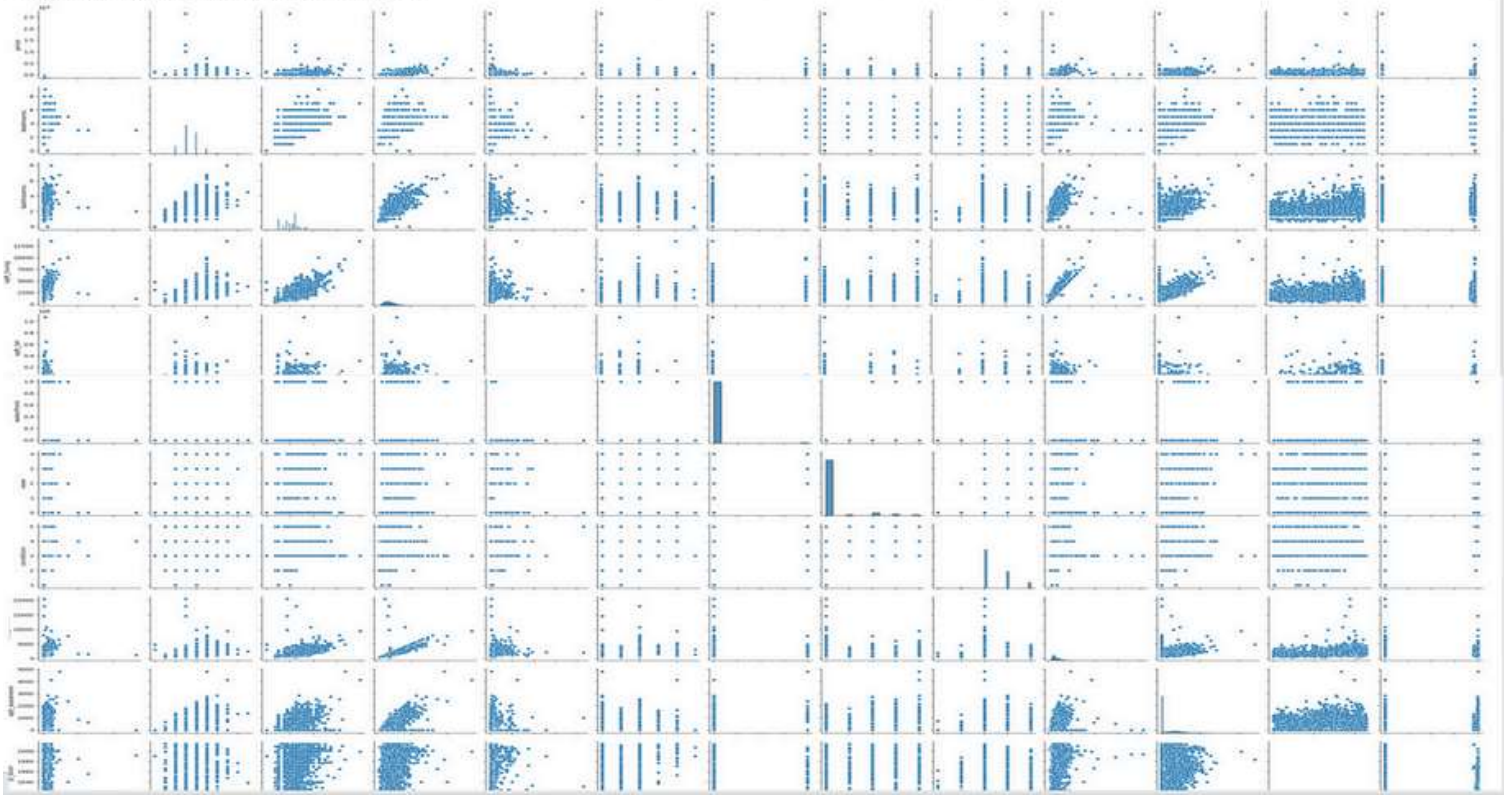
```
data.corr()
```

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	sqft_above	sqft_basement	yr_built	yr_renovated
price	1.000000	0.192793	0.319311	0.419000	0.048293	0.143775	0.131387	0.220758	0.036660	0.311732	0.208591	0.023030	-0.031068
bedrooms	0.192793	1.000000	0.545920	0.593302	0.068796	0.177895	-0.003483	0.111028	0.025080	0.429573	0.334165	0.141070	-0.061082
bathrooms	0.319311	0.545920	1.000000	0.760145	0.107251	0.486428	0.076232	0.211960	-0.119994	0.612828	0.298020	0.461237	-0.215886
sqft_living	0.419000	0.593302	0.760145	1.000000	0.209090	0.343607	0.118055	0.310315	-0.061593	0.772572	0.446905	0.284979	-0.123501
sqft_lot	0.048293	0.068796	0.107251	0.209090	1.000000	0.003832	0.017283	0.074024	0.000406	0.200370	0.034200	0.050154	-0.022048
floors	0.143775	0.177895	0.486428	0.343607	0.003832	1.000000	0.022024	0.031211	-0.275013	0.460266	-0.255510	0.465032	-0.233996
waterfront	0.131387	-0.003483	0.076232	0.118055	0.017283	0.022024	1.000000	0.360935	0.000352	0.068900	0.097501	-0.024839	0.008625
view	0.220758	0.111028	0.211960	0.310315	0.074024	0.031211	0.360935	1.000000	0.063077	0.150515	0.321602	-0.065374	0.022967
condition	0.036660	0.025080	-0.119994	-0.061593	0.000406	-0.275013	0.000352	0.063077	1.000000	-0.168267	0.200632	-0.398093	-0.186818
sqft_above	0.311732	0.429573	0.612828	0.772572	0.200370	0.460266	0.068900	0.150515	-0.168267	1.000000	-0.034381	0.369645	-0.143899
sqft_basement	0.208591	0.334165	0.298020	0.446905	0.034200	-0.255510	0.097501	0.321602	0.200632	-0.034381	1.000000	-0.160952	0.043125
yr_built	0.023030	0.141070	0.461237	0.284979	0.050154	0.465032	-0.024839	-0.065374	-0.398093	0.369645	-0.160952	1.000000	-0.320952
yr_renovated	-0.031068	-0.061082	-0.215886	-0.123501	-0.022048	-0.233996	0.008625	0.022967	-0.186818	-0.143899	0.043125	-0.320952	1.000000

```
sns.pairplot(data)
```

```
<seaborn.axisgrid.PairGrid at 0x1e83b58e7c0>
```

PAIRPLOT



Overall Insights Obtained from Analysis

Based on the EDA, several key insights are typically obtained:

Price Drivers: Identifying which variables (e.g., location, size, number of bedrooms) most significantly impact housing prices.
Trends and Patterns: Uncovering trends such as price fluctuations over time or differences across neighborhoods.
Segment Characteristics: Understanding the characteristics of different property segments or clusters.
Recommendations: Providing actionable recommendations for stakeholders, such as pricing strategies, investment opportunities, or market positioning.
These insights help in making data-driven decisions and understanding market dynamics.

Conclusion

In conclusion, the exploratory data analysis of the housing dataset has provided valuable insights into the factors affecting housing prices and the overall market trends. By addressing data quality issues, deriving meaningful metrics, and performing detailed analyses, we have gained a deeper understanding of the housing market. These findings offer practical implications for real estate professionals and potential buyers, enabling more informed decisions and strategic planning.