# **Business Understanding:**

1. **Problem Identification:**

The main issue in my case is to predict housing prices in California using various features like age of the house, ocean\_proximity, Population etc.

1. **Objectives:**

* To develop a model that should accurately predict the median house value based on other variables by multiple linear regression.
* To understand the housing price for real estate investments and urban planning.
* To evaluate the model’s performance using R-squared and MSE and understand the impact of each feature on housing prices.

1. **DataDictionary:**

| **Variables** | **Description** | **DataType** |
| --- | --- | --- |
| Longitude | Longitude of the location | Numeric |
| Latitude | Latitude of the location | Numeric |
| Housing\_median\_age | Median age of the house | Numeric |
| Total\_rooms | Total number of rooms in the house | Numeric |
| Total\_bedrooms | Total number of bedrooms in the house | Numeric |
| Population | Population in the block | Numeric |
| Households | Number of household in the block | Numeric |
| Median\_income | Median income of the block | Numeric |
| Ocean\_proximity | Proximity to the ocean | Categorical |
| Median\_house\_value | Median house value | Numerical(target variable) |

2. **Data Description :**

* Data Exploration: It has 20433 entries and 10 columns with featuring ranging from geographic, demographic, economic indicators.
* No missing Values

3) **Data Cleaning:**

## Missing value analysis : no missing values.

print(dataset.isnull().sum())

longitude 0

latitude 0

housing\_median\_age 0

total\_rooms 0

total\_bedrooms 0

population 0

households 0

median\_income 0

ocean\_proximity 0

median\_house\_value 0

dtype: int64

## Histogram and correlation analysis :

import pandas as pd

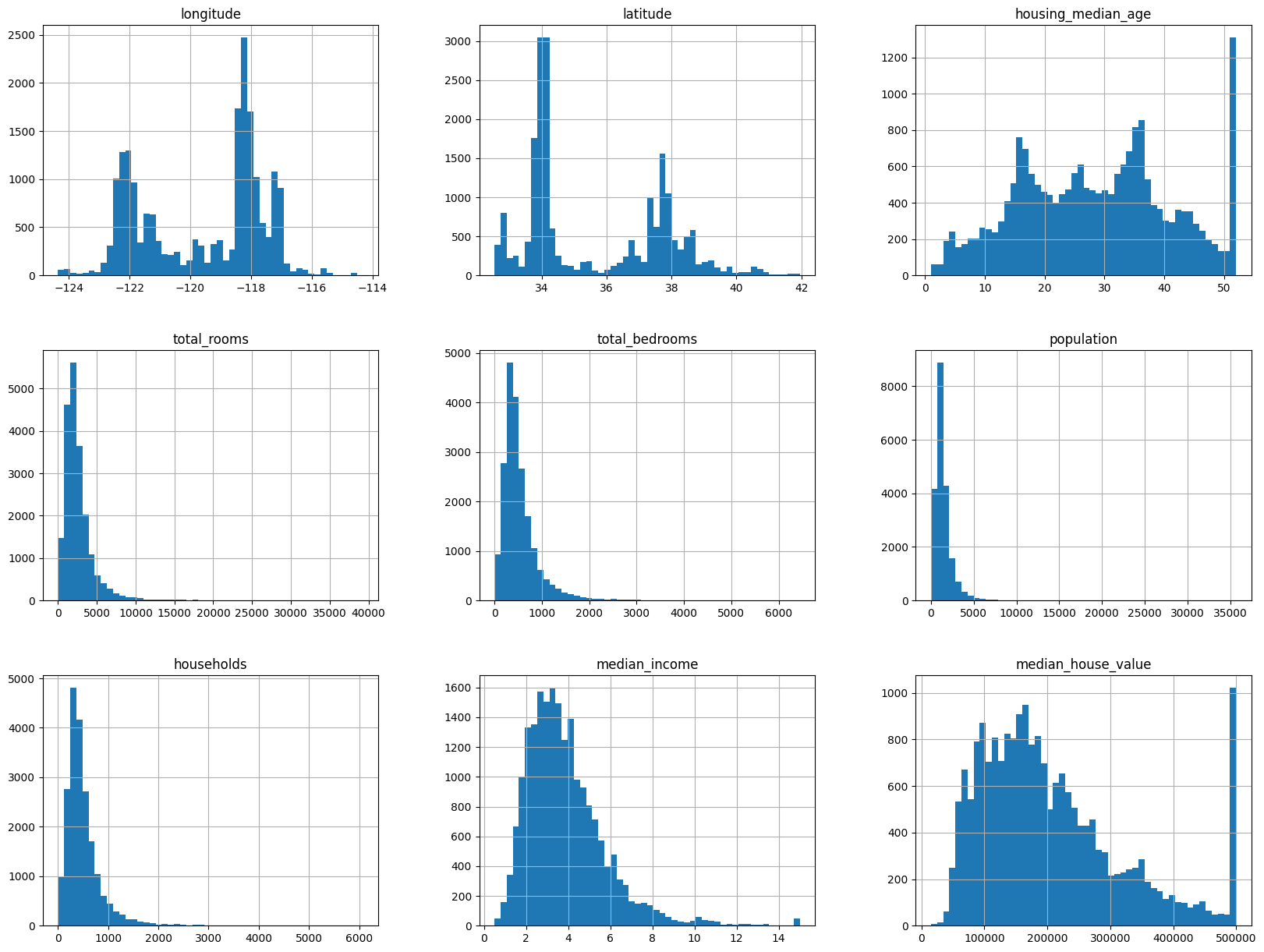
import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

dataset.hist(bins=50, figsize=(20, 15))

plt.show()



Correlation:

numerical\_df = df.select\_dtypes(include=['float64', 'int64'])

# Compute the correlation matrix

correlation\_matrix = numerical\_df.corr()

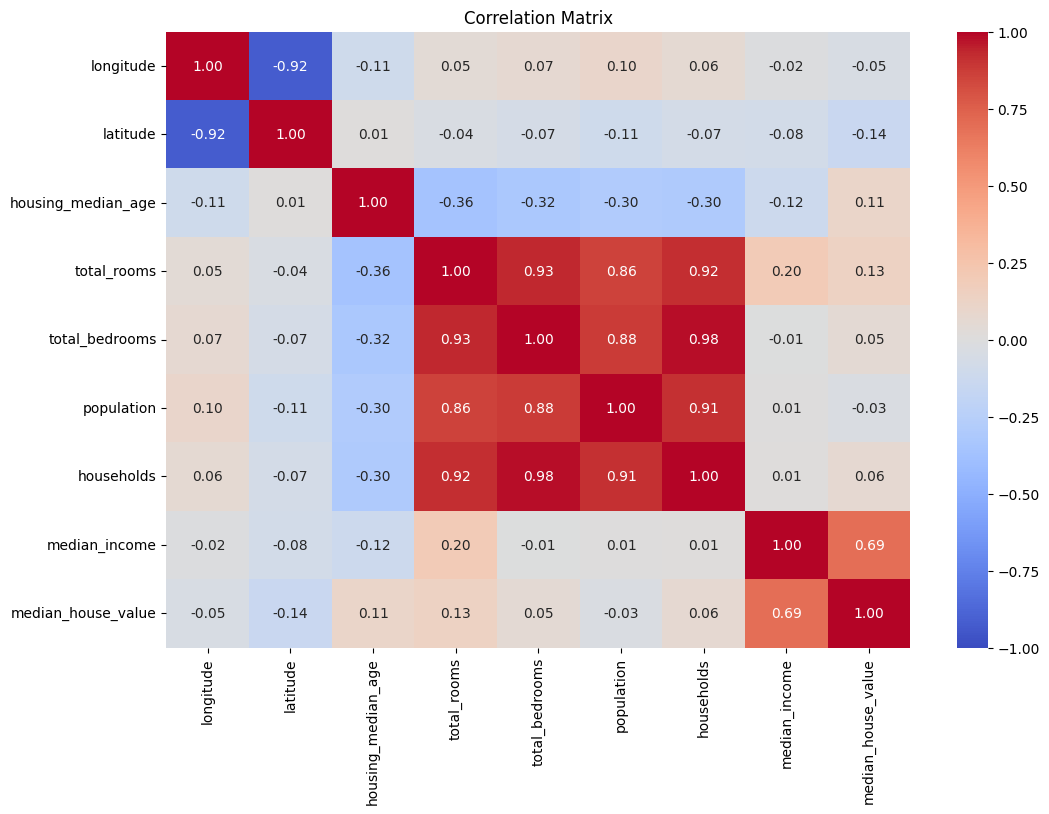
# Plot the heatmap

plt.figure(figsize=(12, 8))

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt='.2f', vmin=-1, vmax=1)

plt.title('Correlation Matrix')

plt.show()



Insights: Median house value is highly correlated by median\_income

4. **Modeling(multiple LInear Regression)**

1. Model selection and Assumptions:
2. Model Output:

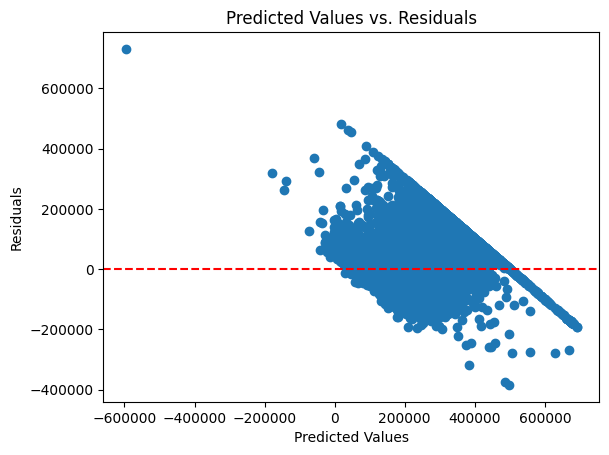


1. Algorithm for multiple regression:

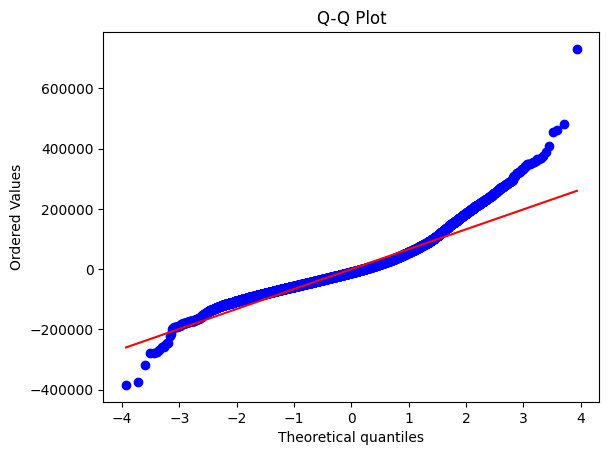
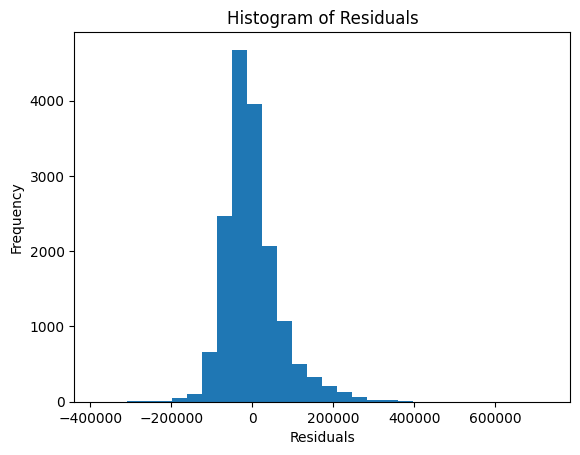


**Model 1: Full Model Multiple linear regression and Assumptions**

1. **Linearity :** Data should be linear relationship between y and each X variable



Insight: It is not linear, So that i can use polynomial regression.

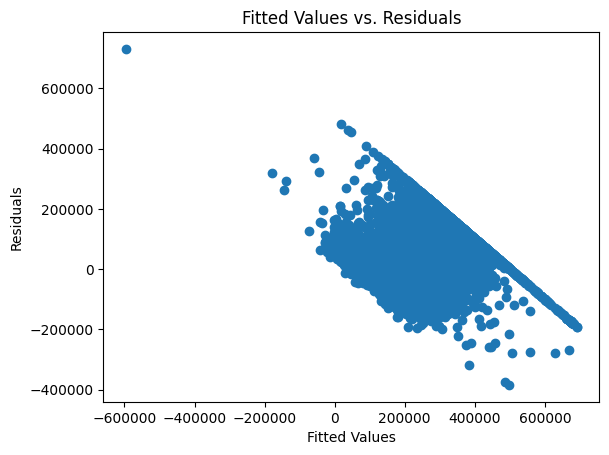
1. **Normality : Normal distribution of error** 

Insights: It follow normality

1. **No autocorrelation**: We don’t want to see any kind of patter in our data. If there it indicates some rows are affecting other rows. I tested through durbin watson it shows no autocorrelation.

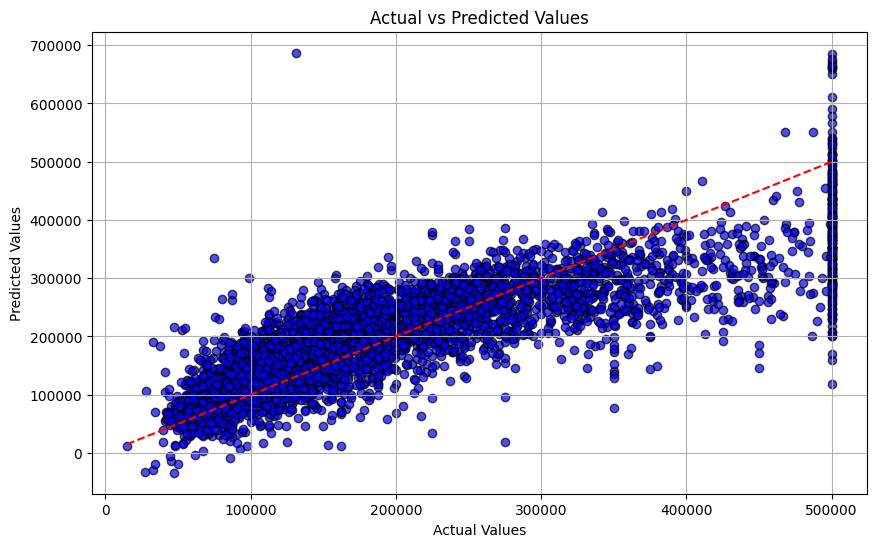
Durbin-Watson statistic: 1.98.

1. **Homoscedasticity:** Same Variance which means variance is dependent on independent variables.



Insights: It is heteroscedasticity. Because housing prices are influenced by some independent variable.

**Model 1: multiple linear regression output:**

****

Mean Absolute Error: 49964.01

Mean Squared Error: 4651447496.46

R-squared: 0.65

**Model Parameters:**

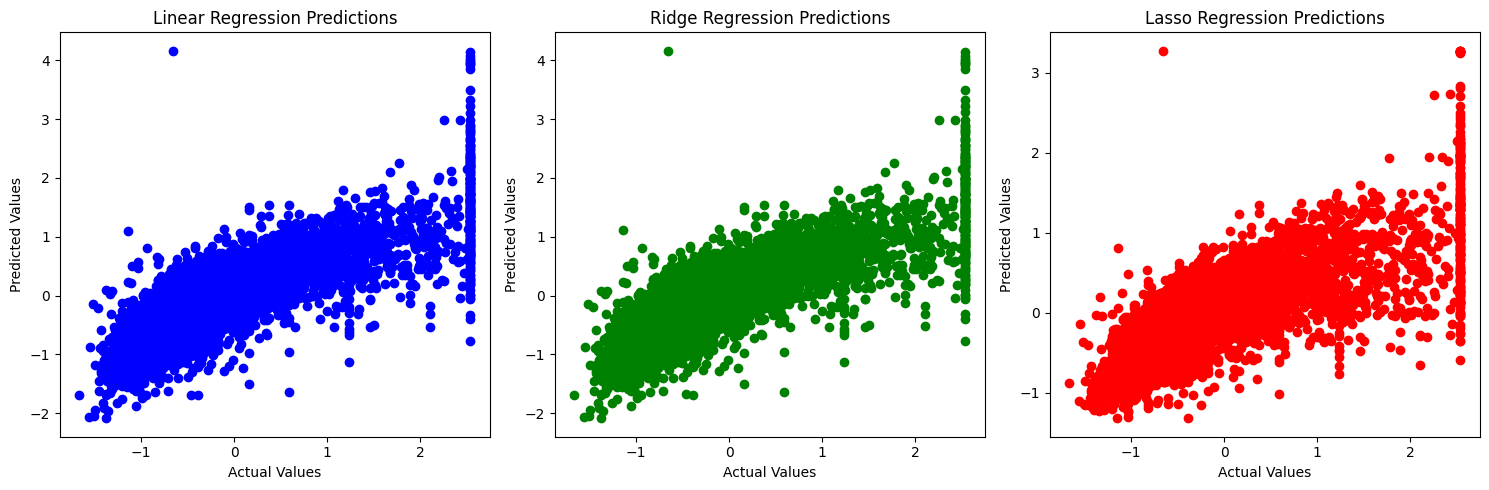
β0​ is the intercept(constant) = -2333801.822611121

Coefficient = [-2.58e+04, -6.41e+04 , 1.42e+05 ,-2.98e+04,

-2.78e+04, -2.63e+04, 1.09e+03, -5.97e+00,

1.01e+02,-3.58e+01, 4.23e+01, 3.91e+04]

| **Model** | **Train Score** | **Test Score** | **R² Score (Test)** |
| --- | --- | --- | --- |
| Linear Regression | 0.6447 | 0.6526 | 0.6526 |
| Ridge Regression | 0.6448 | 0.6526 | 0.6526 |
| Lasso Regression | 0.5673 | 0.5723 | 0.5723 |



**Model 2: PCA output : No of principle component - 6**

Mean Squared Error: 5189374170.962136

R-squared: 0.6205261011060057

**Model 3: Polynomial Regression Output. It shows overfitting.**

Training R-squared: 0.71

Testing R-squared: -648684536192850944.00

Training MSE: 3780200909.99

Testing MSE: 8870878305551385133943095296.00

**Model 4: Ridge and Lasso Regression for regularizing**

Ridge Training R-squared: 0.71

Ridge Testing R-squared: 0.71

Ridge Training MSE: 3780376633.85

Ridge Testing MSE: 3979819736.25

Lasso Training R-squared: 0.71

Lasso Testing R-squared: 0.70

Lasso Training Adjusted R-squared: 0.72

Lasso Testing Adjusted R-squared: 0.70

Lasso Training MSE: 3784630141.11

Lasso Testing MSE: 3981887068.99

Lasso Training MAE: 43794.84

Lasso Testing MAE: 45047.63

**Overall Models:**

| **Model** | **Features** | **R2 value** | **MSE** | **MAE** | **Remark** |
| --- | --- | --- | --- | --- | --- |
| **Full model** | x1…..x9 | 0.65 | 4651447496.46 | 49964.01 | It's a **good baseline model**. |
| **PCA model** | x1….x6 | 0.62 | 5189374170.96 | 52178.5 | This model is **slightly less effective** than the full model, likely due to loss of some information in the PCA transformation. |
| **Polynomial** | X1…. x9 | 0.71 - train  -6486.0: test | 3780200909.99  8870878305513 | 43788.39  15311717384 | This model exhibits extreme **overfitting.** |
| **Lasso** | x1….x9 | 0.71 | 378037663.85 |  | Lasso Regression **performs well,** with an R² value of 0.71 and a significantly lower MSE compared to the full model. |
| **Ridge** | x1….x9 | 0.70 | 398188706.99 | 45047 | **Ridge Regression also performs** well, with an R² value of 0.70, a lower MSE compared to the full model, and a relatively low MAE. |

**Model Interpretation from Business Point of view:**

Lasso is perfect among others. It explains approximately 71% of variance in housing prices on both training and testing data indicating a good fit. The consistency between training and testing r-squared values suggests the model is not over fitting.

After that Ridge and Full model is working well Because 65% of the variance in housing price, indicating a better fit than PCA and Lasso.