Minimalist

Housing Price Prediction

Machine Learning Mini Project
by Raditya Erlang Arkananta

About Me

Raditya Erlang Arkananta

Experience

- **PT Hekikai Indonesia - QA QC Japanese Translator** **2024 - 2025**
- **PT Indonesia Indicator - HR Officer Staff** **2024**

Education

- **Institute Technology of Sepuluh Nopember** **2019 - 2023**
  Bachelor of Science in Industrial Engineering
- **Dibimbing Data Science** **2024 - 2025**
  Currently Learning Data Scientist Skills

Table of Contents

01
Business Understanding & Goals

02
Data Understanding

03
Data Pre-Processing

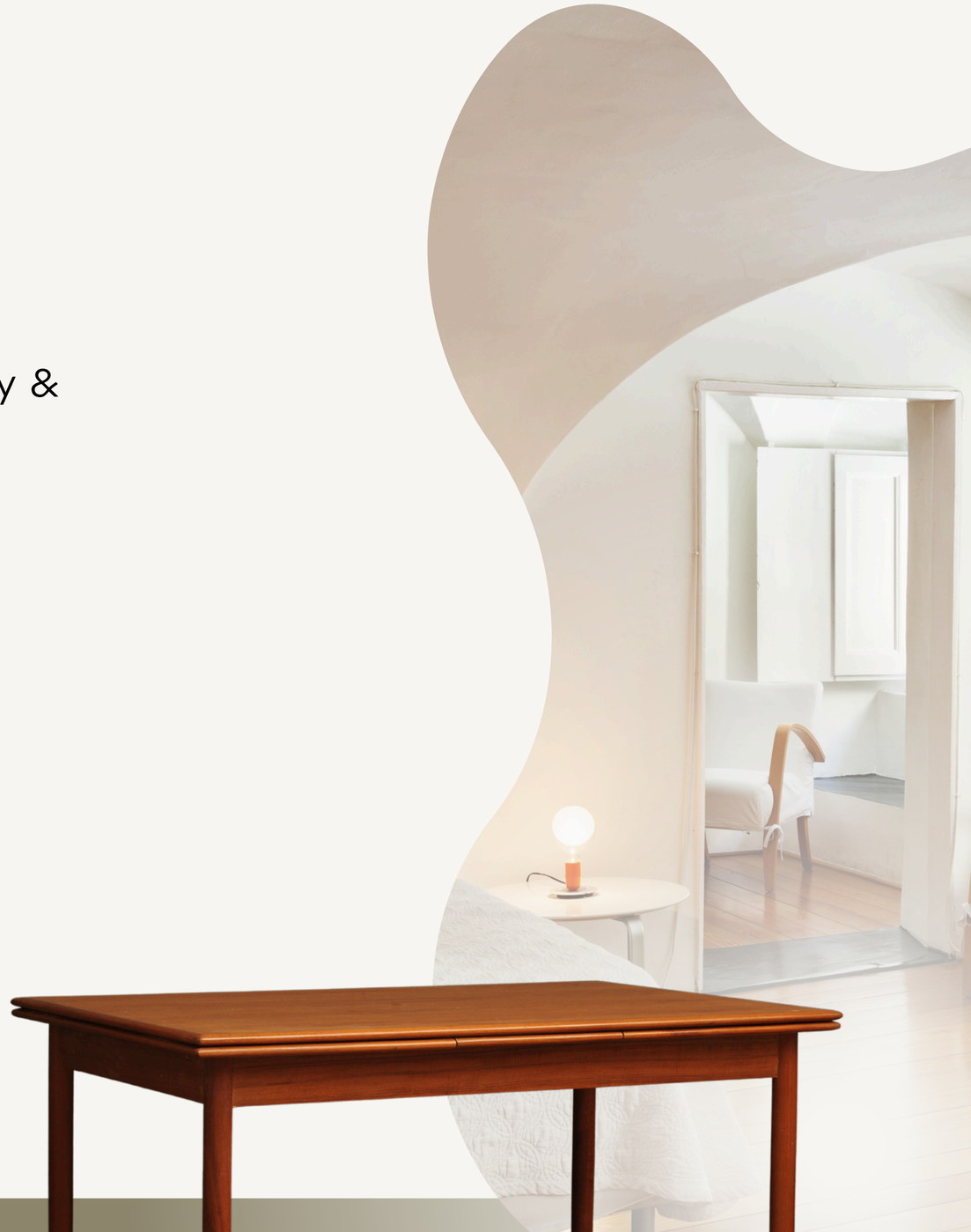
04
Exploratory Data Analysis

05
Multicolinearity Study & Feature Engineering

06
Building Machine Learning Models

07
Model Evaluation

08
Insight & Recommendation



Business Understanding & Goals

Background

Housing prices play a crucial role in the economy, influencing both individual financial decisions and broader economic policies, however due to the complex interplay of multiple variables such as location, size, amenities, economic conditions, interest rates, and market demand, predicting housing prices can be challenging, Machine learning techniques enable us to build predictive models that can learn from historical data and generalize to unseen cases.

Problem

Accurately pricing a house is a complex task, often relying on subjective judgment or outdated comparisons. This can lead to overpricing or underpricing, affecting market efficiency and decision-making for buyers, sellers, and investors.

Goal & Objective

The goal of this Project is to develop a Machine Learning Model that is able to accurately predict the sale price of a house based on its features by leveraging historical housing data such as location, size, number of rooms, and other physical or neighborhood attributes

Data Understanding

The dataset used in this project is sourced from the 1990 California Census and is featured in Chapter 2 of Aurélien Géron's Hands-On Machine Learning with Scikit-Learn and TensorFlow. It contains information about housing and demographic data for various districts across California, obtained from Kaggle.

This Dataset Contains :

> Data from 1990 California Census

> 20460 Rows of Data

> 10 Columns/Feature

It should be noted that the house prices in the dataset represent the median values for each census block group, rather than individual house prices.

Data Understanding

This Dataset contains the Following Columns:

1. longitude: A measure of how far west a house is; a higher value is farther west
2. latitude: A measure of how far north a house is; a higher value is farther north
3. housingMedianAge: Median age of a house within a block; a lower number is a newer building
4. totalRooms: Total number of rooms within a block
5. totalBedrooms: Total number of bedrooms within a block
6. population: Total number of people residing within a block
7. households: Total number of households, a group of people residing within a home unit, for a block
8. medianIncome: Median income for households within a block of houses (measured in tens of thousands of US Dollars)
9. medianHouseValue: Median house value for households within a block (measured in US Dollars)
10. oceanProximity: Location of the house w.r.t ocean/sea

Data Pre-Processing

1

Check Duplicate
No Duplicate

2

Handling Missing Value
Filled Missing
Numerical
Value with Median

3

Adjusting Incorrect Data Types
Datatypes Already Correct

4

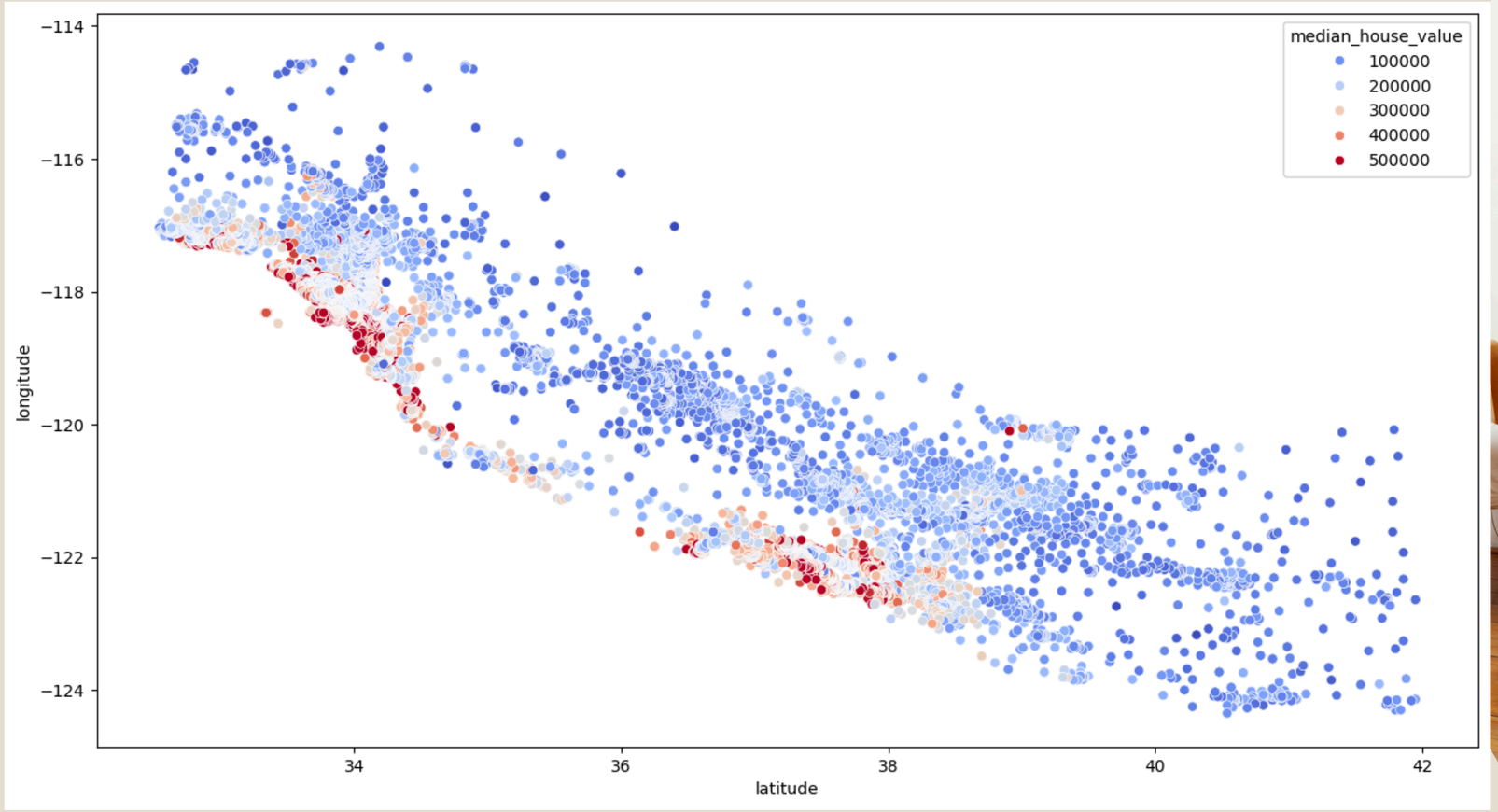
Saving Data to Feature Engineer

Exploratory Data Analysis

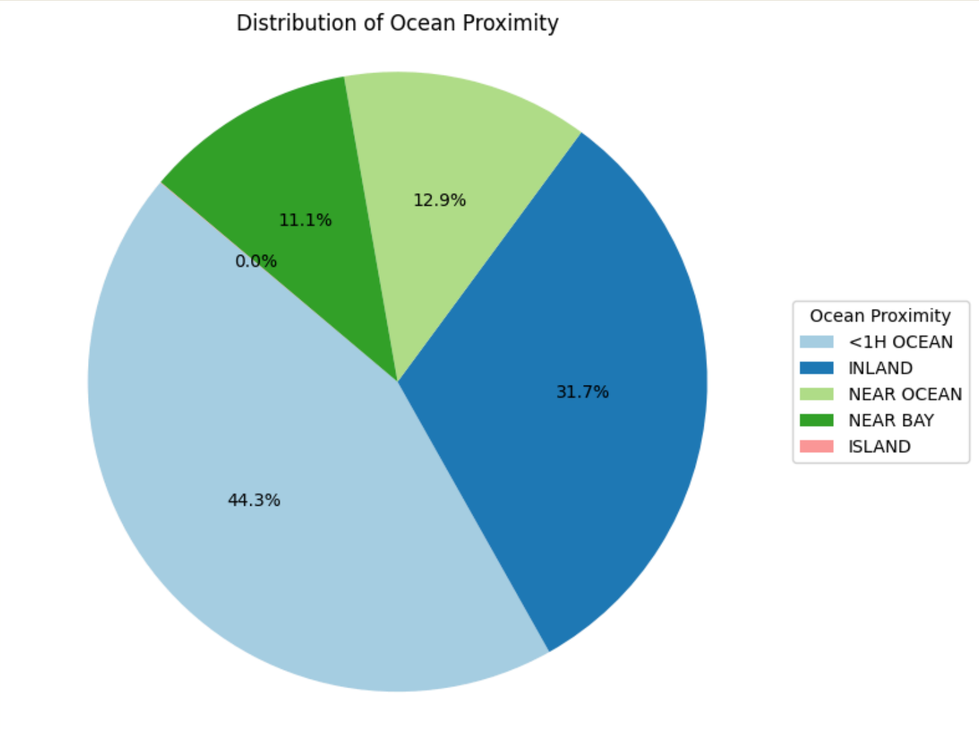
Using latitude and longitude to represent the spatial location of each district, we created a scatter plot.

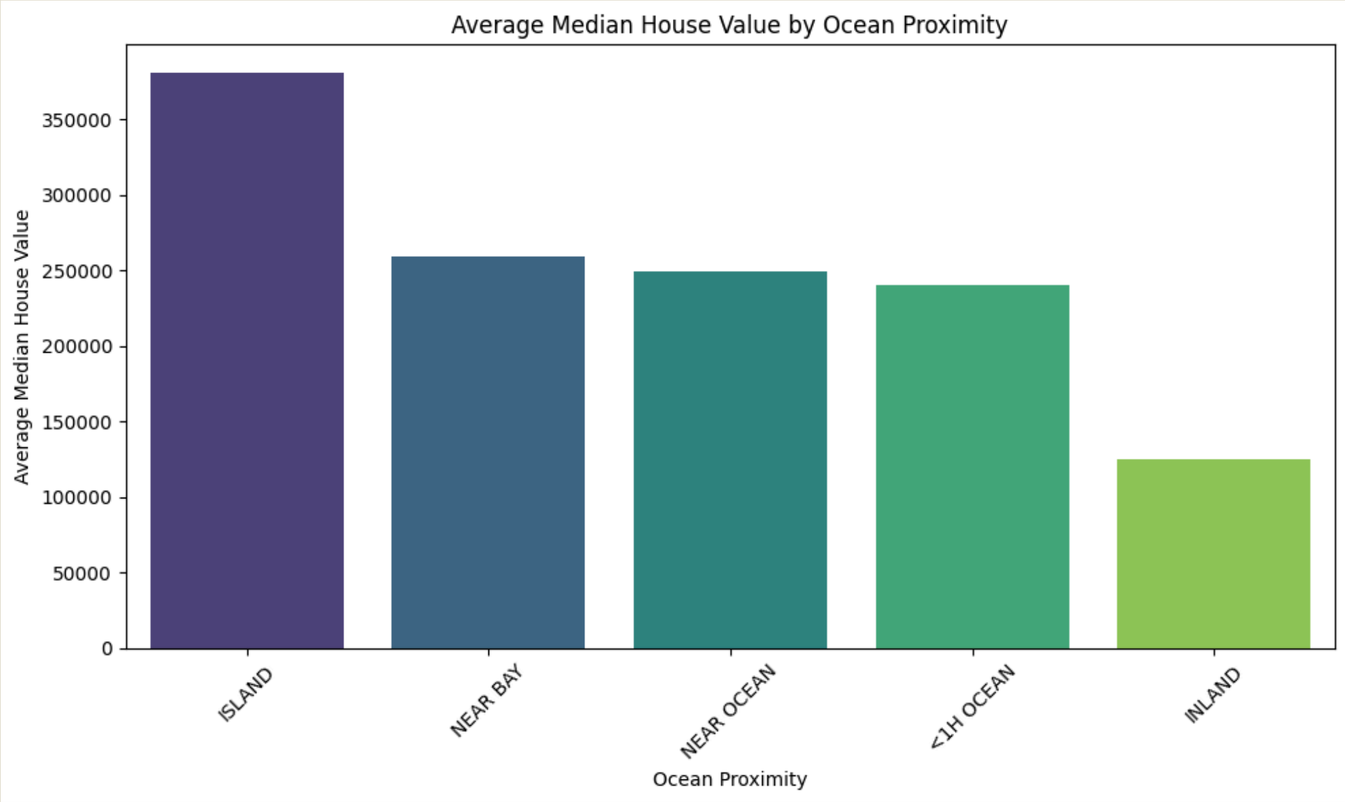
The color of each point represents the median house value, allowing us to identify any regional patterns or trends in housing prices.

- Houses located near the coast tend to have higher prices
- Inland houses are generally more affordable.



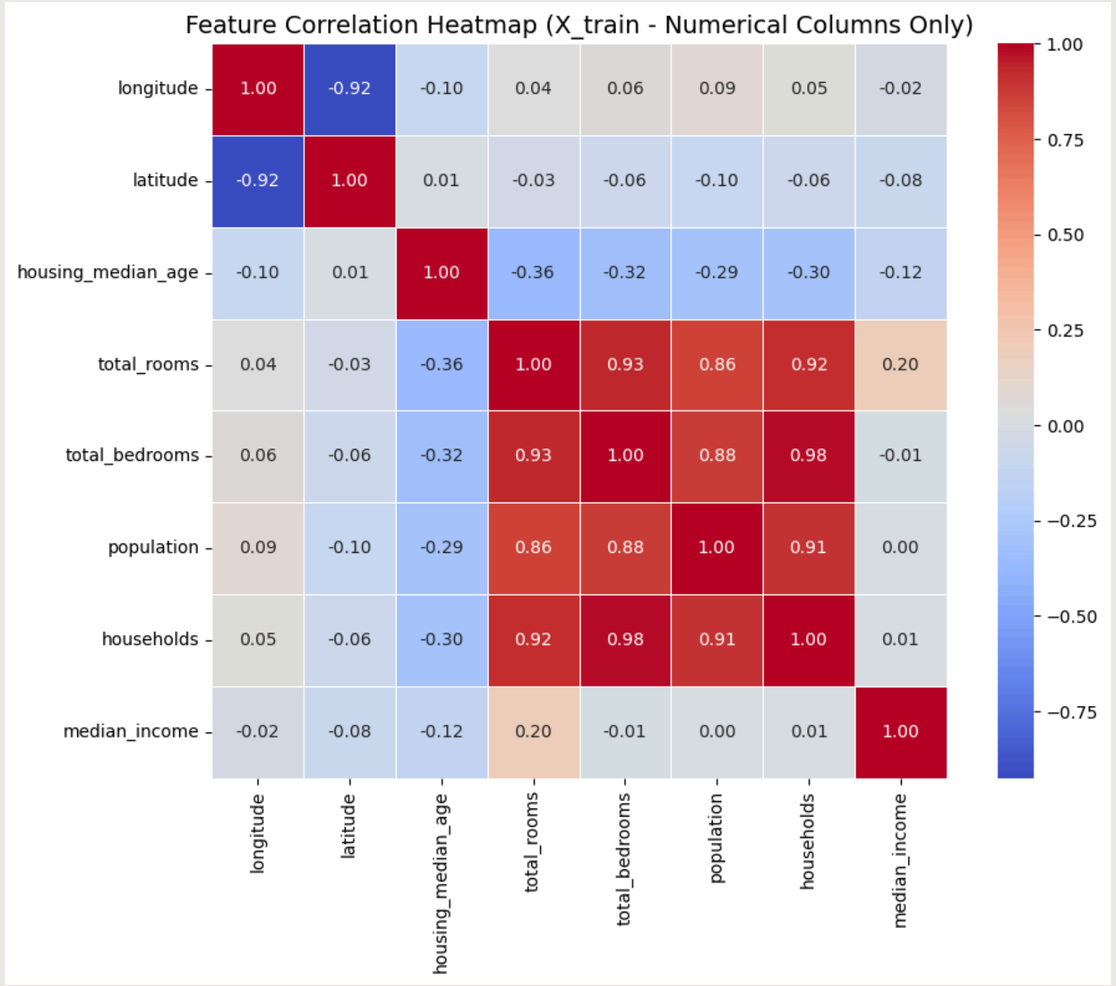
Exploratory Data Analysis





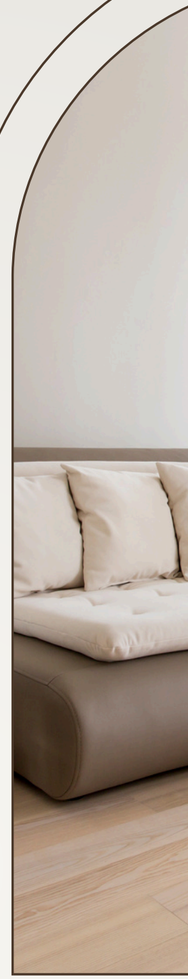
As previously visualized, the average house prices in the categories "NEAR BAY," "NEAR OCEAN," and "<1H OCEAN" are relatively similar, indicating comparable market values in coastal areas. However, there is a clear distinction with "INLAND" properties, which are significantly more affordable, reinforcing the impact of proximity to the ocean on housing prices.

Multicolinearity Study & Feature Engineering

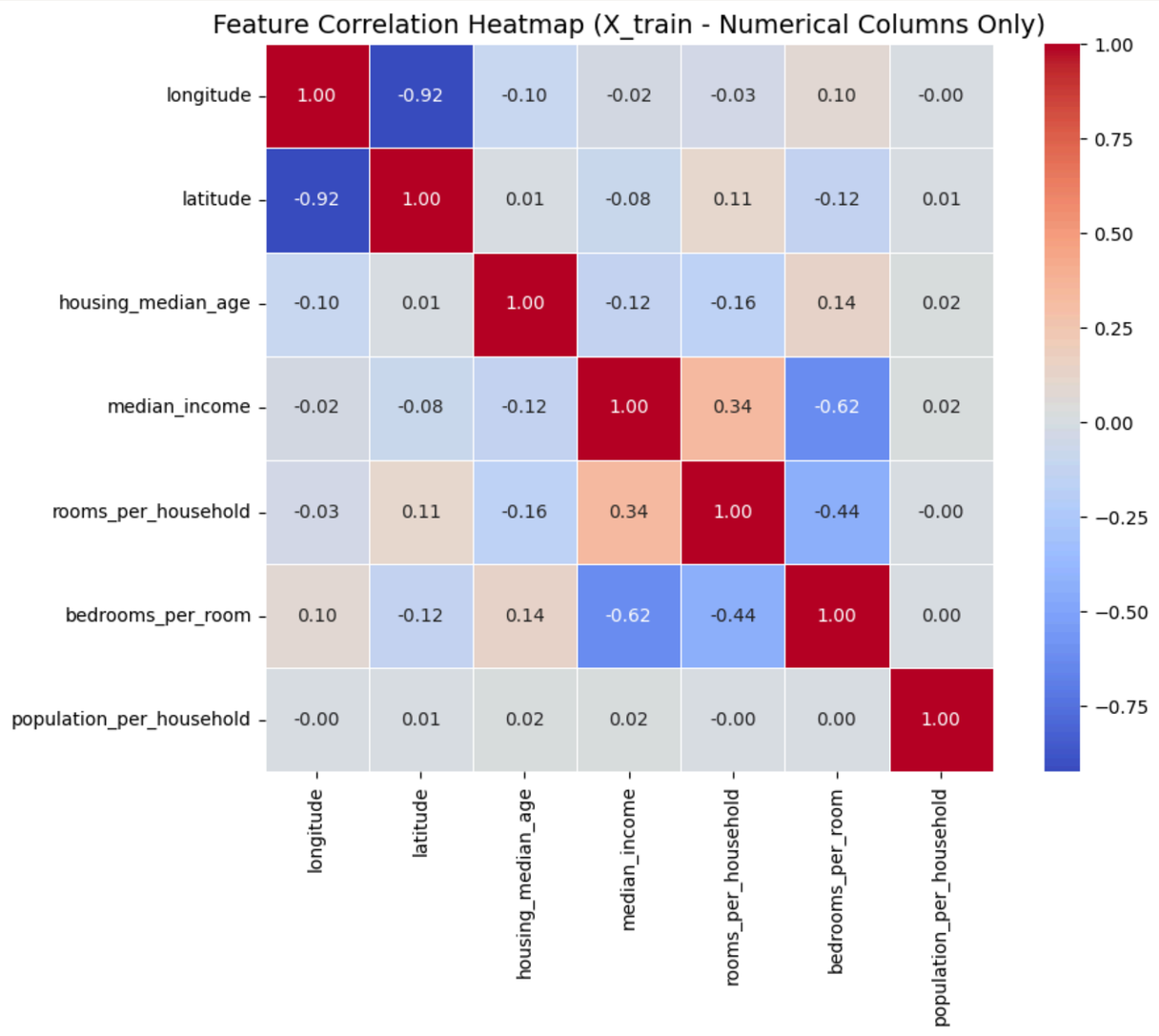


	Feature	VIF_score
0	const	17105.976643
7	households	17.610226
5	total_bedrooms	13.731321
4	total_rooms	10.602362
2	latitude	8.977335
1	longitude	8.848848
6	population	6.217995
8	median_income	1.593887
3	housing_median_age	1.272474

As shown in the VIF scores, several features–such as households, total_bedrooms, and total_rooms–exhibit high multicollinearity, with VIF values well above the common threshold of 10.
We will proceed with feature engineer these features to reduce Multicollinearity.



Feature Engineering



To address multicollinearity, we engineered new ratio-based features

- rooms_per_household was made from total_rooms and households, which captures the average number of rooms per household.
- bedrooms_per_room was derived from total_bedrooms and total_rooms
- population_per_household was created from population and households.

These new features are more informative and reduce redundancy. After creating them, we dropped the original columns–total_bedrooms, households, total_rooms.

I decide to retain both longitude and latitude because together they provide geographical context that is highly relevant to housing prices such as proximity to the coast, urban centers, or specific regions

Scaling & Encoding

Finally, we encode the categorical variables and scale the numerical features to ensure that all input data is in a consistent format and on a comparable scale.
This step is essential for improving model performance, especially for algorithms that are sensitive to feature magnitudes, such as linear regression or gradient boosting.

```
one_hot_encode_cols = ["ocean_proximity"]
# One-Hot Encoding for multi-class categorical columns
X_train = pd.get_dummies(X_train, columns=one_hot_encode_cols, drop_first=False)
```

✓ 0.0s

```
X_train_numeric = X_train.select_dtypes(include='number')

# Initialize the scaler
scaler = StandardScaler()

# Fit and transform the numeric columns
X_train[X_train_numeric.columns] = scaler.fit_transform(X_train_numeric)
```

✓ 0.0s

Machine Learning Model

→

Random Forest

XGBRegressor

GradientBoostingRegressor

Random Forest Regressor

The first model we experiment with is Random Forest Regressor, a powerful ensemble learning method known for its ability to handle non-linear relationships and reduce overfitting by averaging the results of multiple decision trees.

Using this Model, we get:

```
MAE:  32042.97
MSE:  2507224097.53
RMSE: 50072.19
R2:   0.8087
```

The MAE means that the model's predictions deviate from the actual house prices by around $32K, RMSE is 50,072.19, indicating that larger errors are more penalized.

The R^2 score is 0.8087, approximately 81% of the variance is explained.

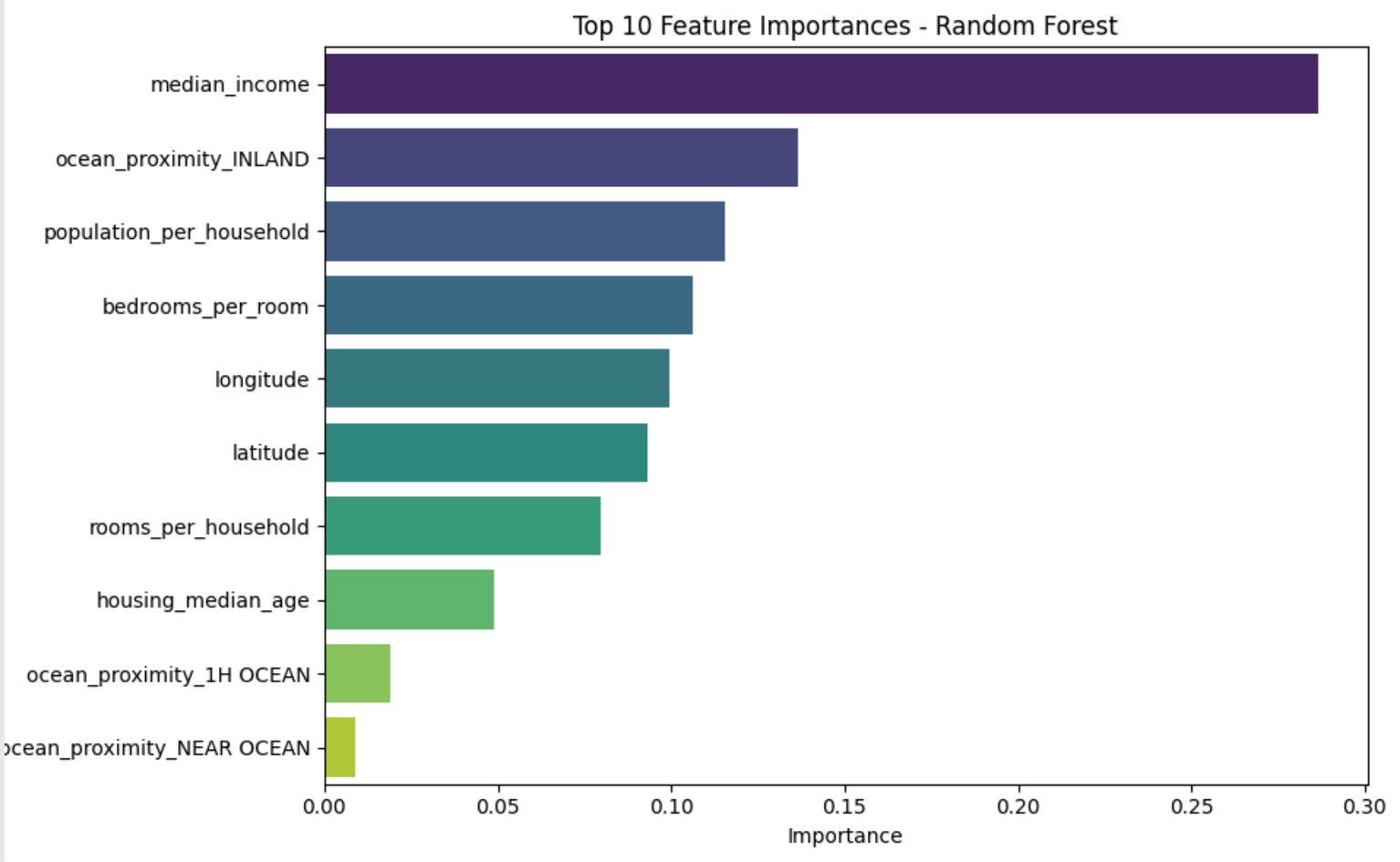
Next, we tune the Hyperparameter according to the following grid and got

```
param_grid = {
    'n_estimators': [100, 200],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2],
    'max_features': ['auto', 'sqrt']
}
```

✓ 0.0s

```
MAE:  31166.31
MSE:  2507224097.53
RMSE: 47561.05
R2:   0.8274
```

Random Forest



According to the Random Forest Model, the Feature that Impacts the Model the most is the Median Income of the Block

	Feature	Importance
3	median_income	0.286493
8	ocean_proximity_INLAND	0.136569
6	population_per_household	0.115367
5	bedrooms_per_room	0.106232
0	longitude	0.099529
1	latitude	0.093314
4	rooms_per_household	0.079554
2	housing_median_age	0.048806
7	ocean_proximity_1H OCEAN	0.019095
11	ocean_proximity_NEAR OCEAN	0.008748

XGBoost

XGBoost Regressor is a powerful and efficient machine learning algorithm based on gradient boosting. It builds an ensemble of decision trees sequentially, where each new tree corrects the errors of the previous ones

```
MAE:  31662.32
MSE:  2507224097.53
RMSE: 48490.82
R2:   0.8206
```

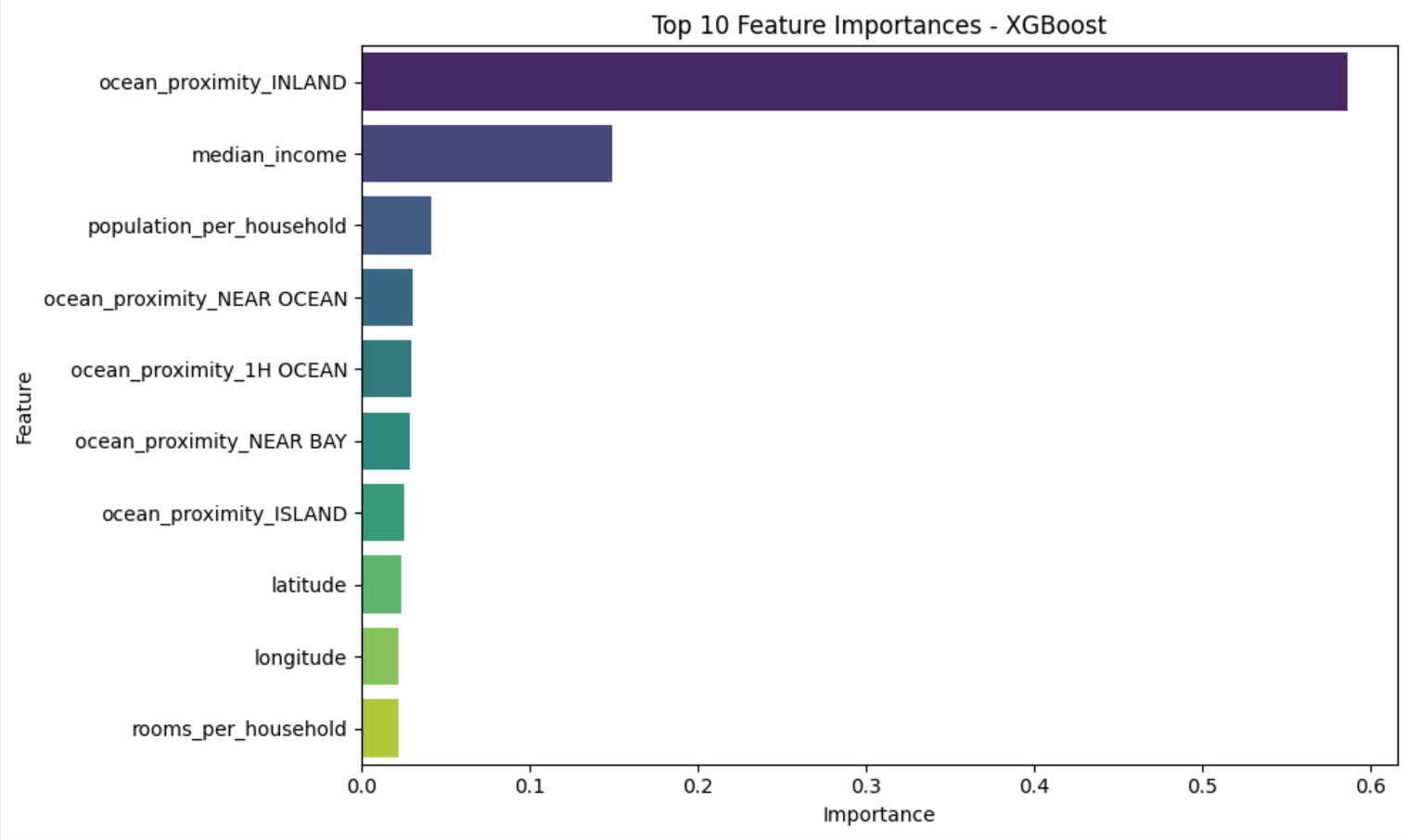
Next, we tune the Hyperparameter according to the following grid and got

```python
# Define parameter grid
param_dist = {
    'n_estimators': [100, 200, 300],
    'learning_rate': [0.01, 0.05, 0.1, 0.2],
    'max_depth': [3, 5, 7, 10],
    'subsample': [0.6, 0.8, 1.0],
    'colsample_bytree': [0.6, 0.8, 1.0],
    'gamma': [0, 1, 5],
    'reg_lambda': [0, 1, 5],
    'reg_alpha': [0, 1, 5]
}
```

✓ 0.0s

```
MAE:  29663.50
MSE:  2507224097.53
RMSE: 46043.20
R2:   0.8382
```

XGBoost



According to the XGRegressor Model, the Feature that Impacts the Model the most is if the House is INLAND

	Feature	Importance
8	ocean_proximity_INLAND	0.587041
3	median_income	0.149440
6	population_per_household	0.041451
11	ocean_proximity_NEAR OCEAN	0.030602
7	ocean_proximity_1H OCEAN	0.029415
10	ocean_proximity_NEAR BAY	0.029041
9	ocean_proximity_ISLAND	0.025567
1	latitude	0.023503
0	longitude	0.022372
4	rooms_per_household	0.022147

GradientBoost

Gradient Boosting Regressor is a machine learning algorithm that builds an ensemble of decision trees sequentially. Each new tree focuses on correcting the errors made by the previous ones by minimizing a loss function

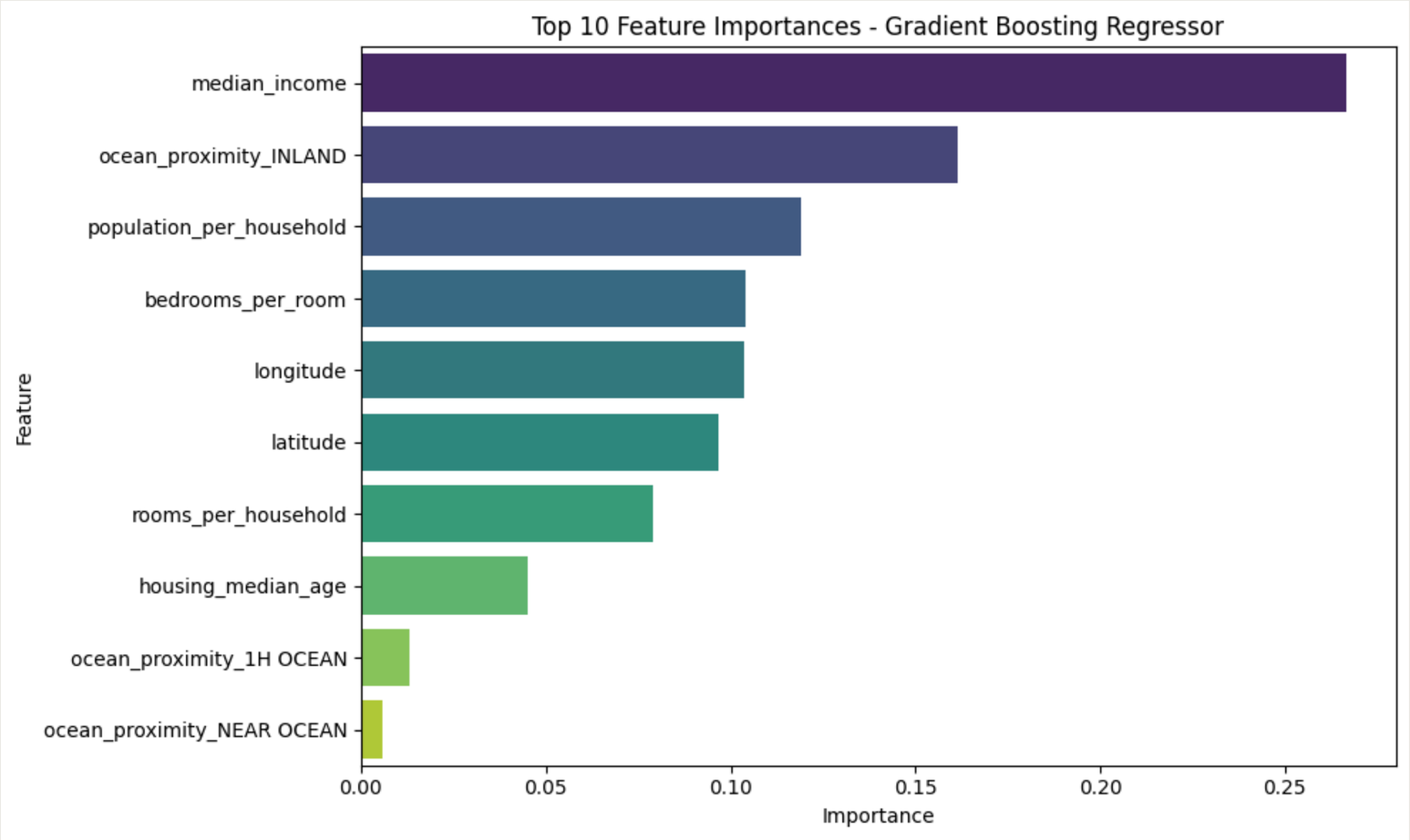
```
MAE:  36854.02
MSE:  2507224097.53
RMSE: 53861.60
R2:   0.7786
```

Next, we tune the Hyperparameter according to the following grid and got

```python
param_dist = {
    'n_estimators': [100, 200, 300],
    'learning_rate': [0.01, 0.05, 0.1, 0.2],
    'max_depth': [3, 5, 7, 10],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'subsample': [0.6, 0.8, 1.0],
    'max_features': ['auto', 'sqrt', 'log2']
}
```

```
MAE:  29330.24
MSE:  2507224097.53
RMSE: 45728.29
R2:   0.8404
```

GradientBoost



According to the Gradient Boost Regressor Model, the Feature that Impacts the Model the most is the Median Income of the Block

	Feature	Importance
3	median_income	0.266747
8	ocean_proximity_INLAND	0.161595
6	population_per_household	0.119122
5	bedrooms_per_room	0.104130
0	longitude	0.103728
1	latitude	0.096600
4	rooms_per_household	0.078917
2	housing_median_age	0.045105
7	ocean_proximity_1H OCEAN	0.013238
11	ocean_proximity_NEAR OCEAN	0.005660

Evaluation - Best Model

Model	Metric	Without Hyperparameters	With Hyperparameters
Random Forest	MAE	32042.97	31166.31
	MSE	2507224097.53	2507224097.53
	RMSE	50072.19	47561.05
	R^2	0.8087	0.8274
XGBoost	MAE	31662.32	29663.50
	MSE	2507224097.53	2507224097.53
	RMSE	48490.82	46043.20
	R^2	0.8206	0.8382
GradientBoost	MAE	36854.02	29330.24
	MSE	2507224097.53	2507224097.53
	RMSE	53861.60	45728.29
	R^2	0.7786	0.8404

We consider the R^2 score to be the most important evaluation metric in this case because it reflects how well the model explains the variance in house prices.

Therefore, Gradient Boosting is our preferred model for this.




Insight & Recommendation

Insight

- In predicting house prices the most impactful feature is median_income, indicating that areas with higher income levels tend to have significantly higher house values.
- ocean_proximity_INLAND also shows strong influence (0.162), reinforcing earlier observations that inland houses are generally cheaper compared to coastal ones
- population_per_household and bedrooms_per_room follow next, suggesting that overcrowding and bedroom density are relevant to housing value
- Geographical features like longitude and latitude still carry moderate importance, implying that location-specific factors (possibly proximity to jobs, amenities, or city centers) affect house prices

Recommendation

- Prioritize High-Income Areas for Investment or Marketing, Since median_income is the strongest predictor of house prices, real estate agencies, property developers, or advertisers should focus resources on high-income neighborhoods where housing demand and prices are higher
- The model confirms that inland properties are valued significantly lower. Businesses can:

1. Target inland areas for affordable housing developments.
2. Market inland homes as budget-friendly options to first-time buyers.

- Aim for balanced room-to-bedroom ratios, promoting comfort and livability.

Let’s Connect!

LinkedIn
Erlang_work@yahoo.com
wa.me/6281218900315

Raditya Erlang
Arkananta

Thank You!