

# California Housing Price

Looking to buy or invest in California real estate?  
Our model helps you estimate home values before making decisions.



# Latar Belakang

## Kebutuhan Pokok

Kebutuhan pokok manusia salah satunya papan yakni hak atas kepemilikan properti seperti rumah. Harga rumah sangat bervariasi di setiap negara, dan saat ini Amerika Serikat termasuk salah satu negara dengan harga properti tertinggi di dunia.

## Kepadatan Penduduk

California, sebagai salah satu negara bagian di Amerika Serikat, memiliki populasi lebih dari 39,2 juta jiwa dan luas wilayah sekitar  $423.970 \text{ km}^2$  (163.696 mil persegi).

## Dinamika Pasar

Dalam upaya memahami dinamika pasar properti, salah satu lembaga riset yaitu Miller Samuel Inc., telah melakukan pendataan selama beberapa tahun terakhir. Lembaga ini secara khusus mengumpulkan informasi penjualan rumah mewah, termasuk rumah keluarga tunggal dan kondominium dengan nilai transaksi di atas USD 50 juta, khususnya di wilayah Amerika Serikat.

# Stakeholder

yang perlu menentukan harga jual atau beli yang kompetitif



## Developer

Developer bisa fokus membangun di area dengan potensi return yang tinggi, bukan hanya berdasarkan intuisi atau tren sementara.

## Agen Real Estate

Agen real estate dapat menunjukkan prediksi harga ke klien untuk mendukung argumen harga jual dan membangun kepercayaan.

# Masalah

Harga rumah di California sangat berfluktuasi dan sulit diprediksi karena banyak faktor (lokasi, populasi, pendapatan, dan infrastruktur).

## Kepadatan Penduduk

Dalam proyek ini, seorang pengembang properti di wilayah California menghadapi tantangan dalam meningkatkan pendapatan dan keuntungan perusahaan.

Salah satu penyebab utama permasalahan tersebut adalah strategi pemasaran yang kurang efektif. Pengembang tersebut menerapkan pendekatan seragam dengan membangun seluruh tipe rumah tanpa mempertimbangkan preferensi pasar dan daya beli masyarakat.

# Solusi

Untuk mengatasi permasalahan ini, pengembang perlu melakukan analisis dan prediksi harga rumah secara lebih akurat.



## Domain Business

Menetapkan harga jual yang kompetitif dan tetap sesuai kebutuhan finansial masyarakat setempat.



## Machine Learning

pihak Developer Properti memiliki sistem berbasis aplikasi yang mampu memprediksi harga rumah secara otomatis berdasarkan data demografis.



## Tujuan

Dengan strategi harga yang tepat sasaran, diharapkan perusahaan memperoleh keuntungan yang optimal dari setiap penjualan rumah.

# Pendekatan Analitis

- 1 MENGIDENTIFIKASI POLA DARI MASING-MASING FITUR YANG TERSEDIA SERTA MENGENALI PERBEDAAN KARAKTERISTIK ANTAR WILAYAH PERUMAHAN.**
- 2 AKAN DIKEMBANGKAN SEBUAH APLIKASI BERBASIS MODEL REGRESI YANG DIRANCANG UNTUK MEMBANTU DEVELOPER PROPERTI DALAM MEMPREDIKSI HARGA RUMAH YANG AKAN DIBANGUN.**
- 3 TUJUANNYA UNTUK MENGESTIMASI ANGGARAN PEMBANGUNAN SECARA OPTIMAL, SEHINGGA MEMPEROLEH KEUNTUNGAN MAKSIMAL DENGAN PENAWARAN HARGA YANG KOMPETITIF DI PASAR.**

# Attribute Information

**14,448 Rows  
&  
10 Columns**

longitude

population

latitude

households

housing\_median\_age

median\_income

total\_rooms

ocean\_proximity

total\_bedrooms

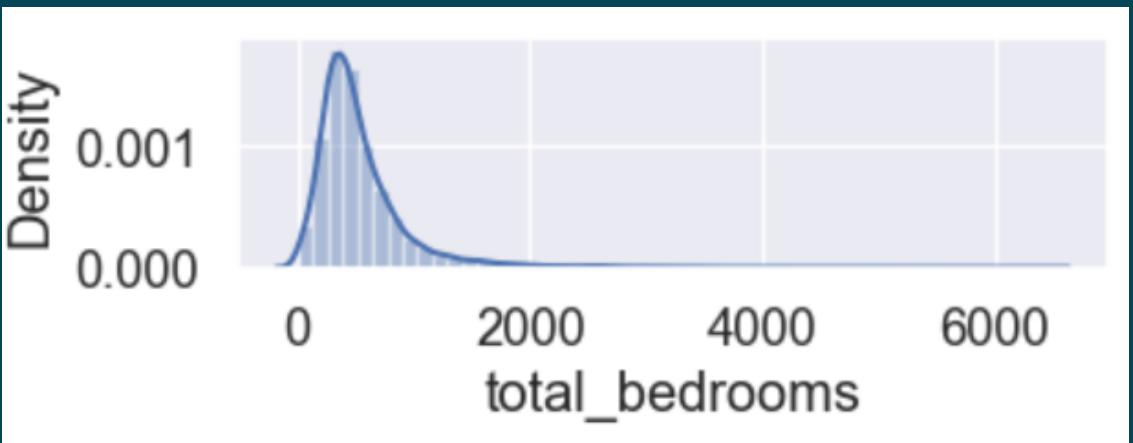
median\_house\_value

# Missing Value

Total bedrooms

137

KDE Plot

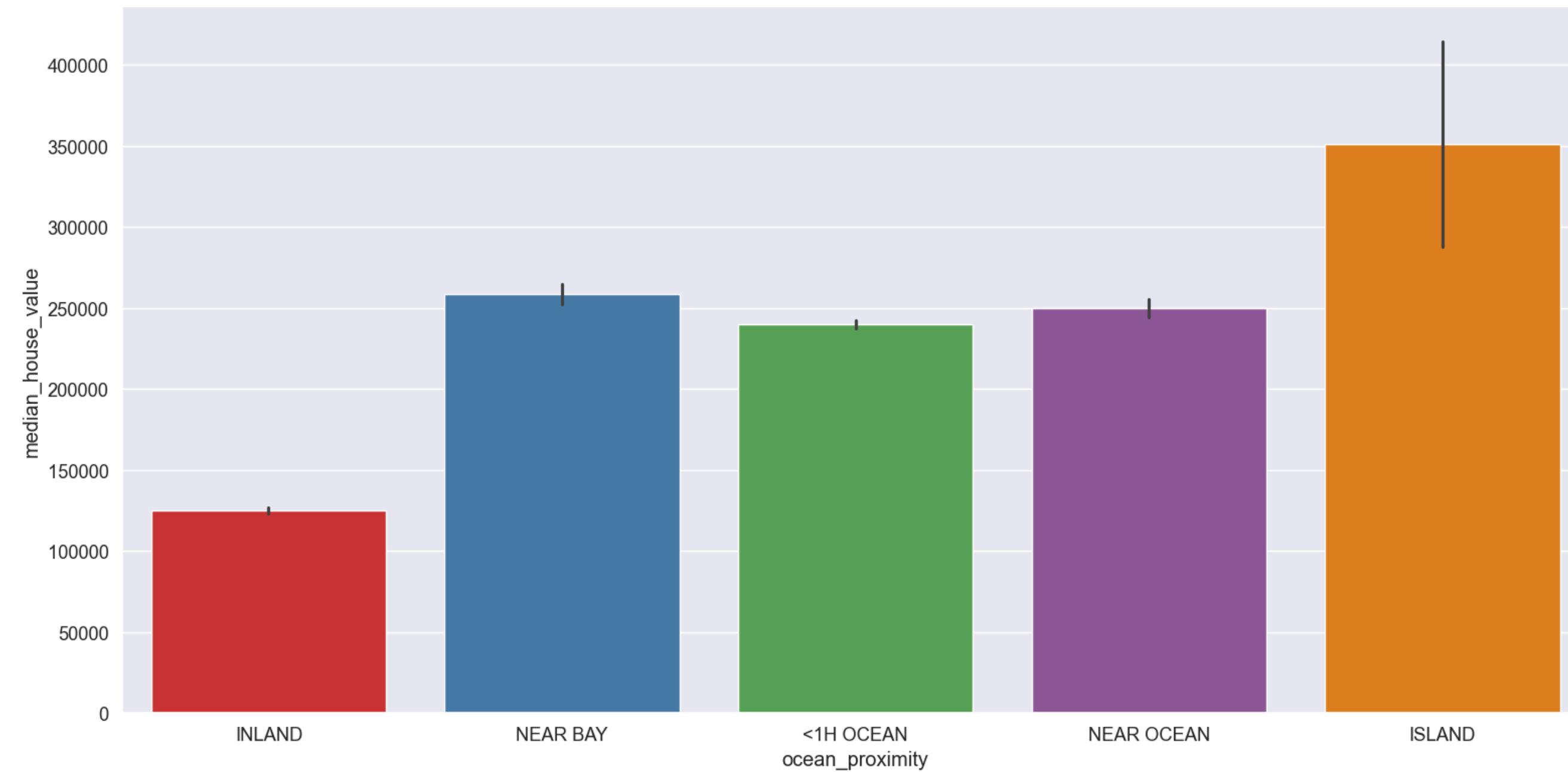


Solusi

Karena distribusi total\_bedrooms bersifat skewed dan terdapat outlier, pendekatan median dipilih untuk mengisi nilai yang hilang agar tidak mengganggu keseimbangan distribusi dan meningkatkan keakuratan model.

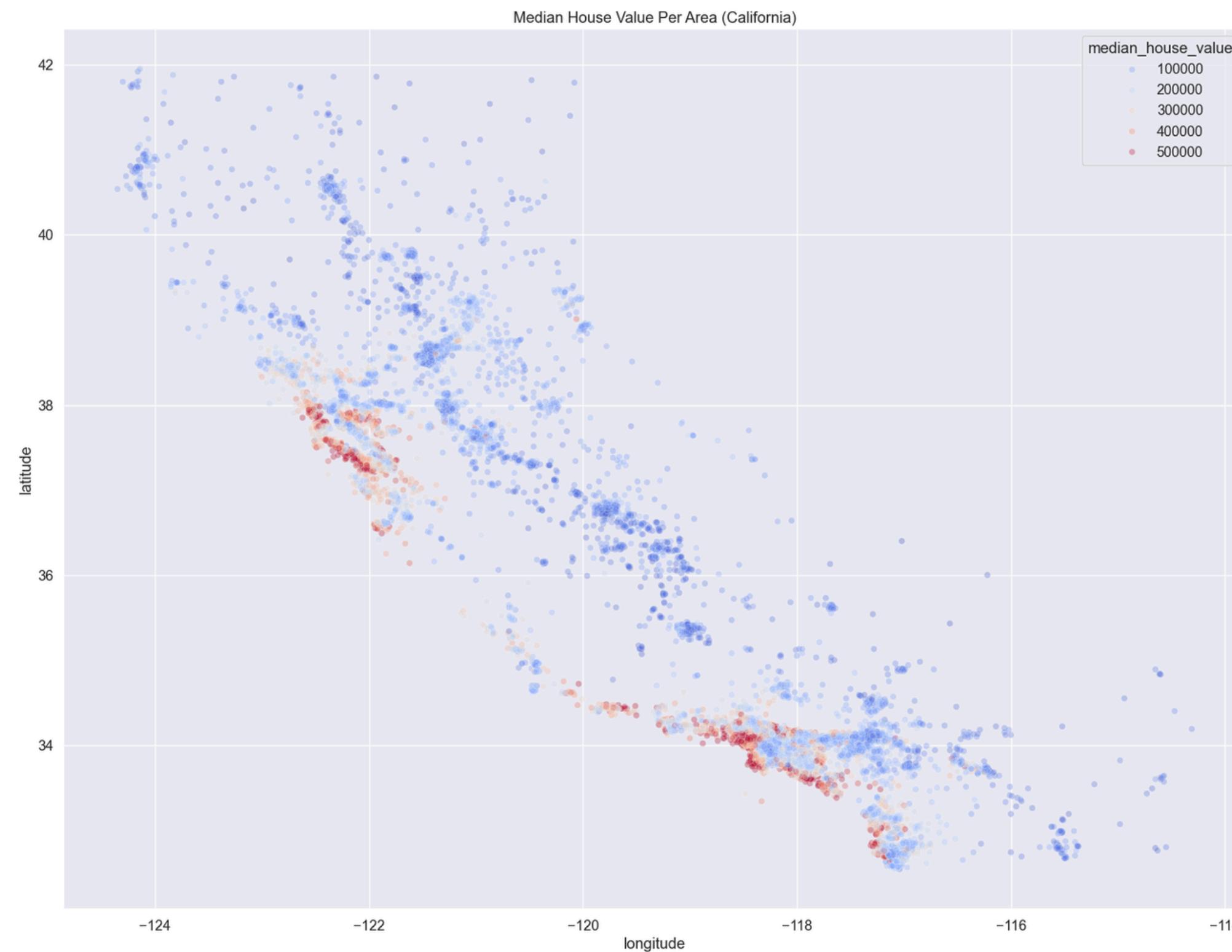
- Distribusi sangat miring ke kanan (right-skewed).
- Sebagian besar nilai berada di bawah 1.000 total bedrooms.
- Ada outlier (nilai ekstrem) di atas 4.000 bahkan hingga lebih dari 6.000.

# EDA



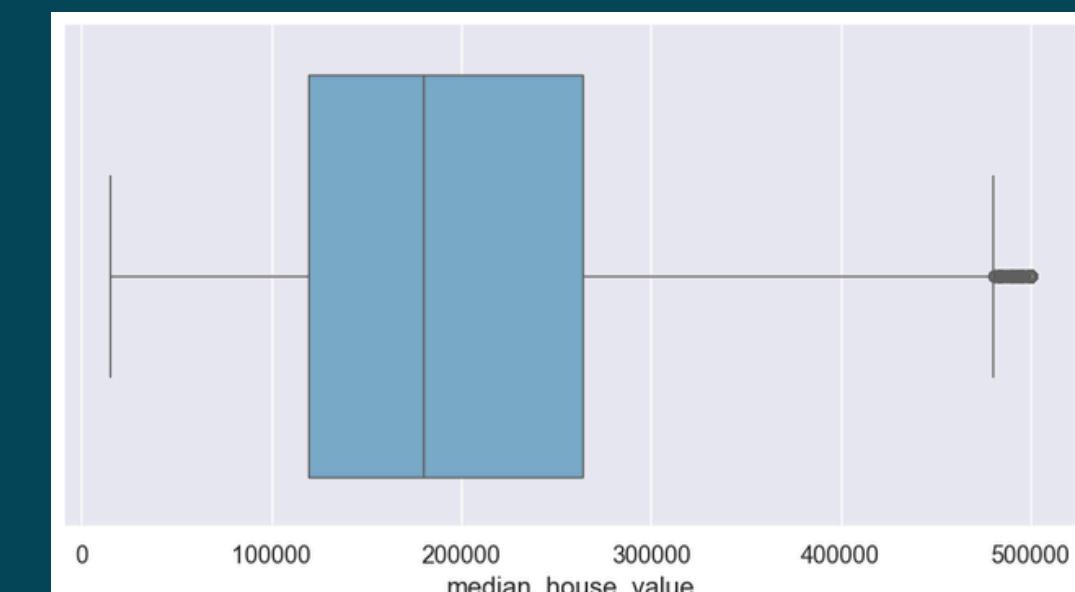
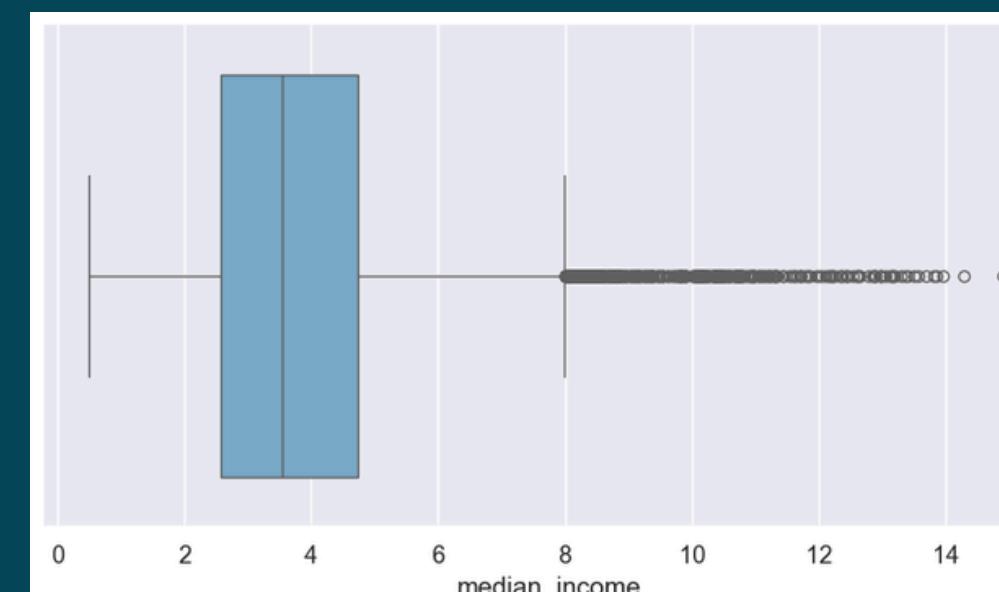
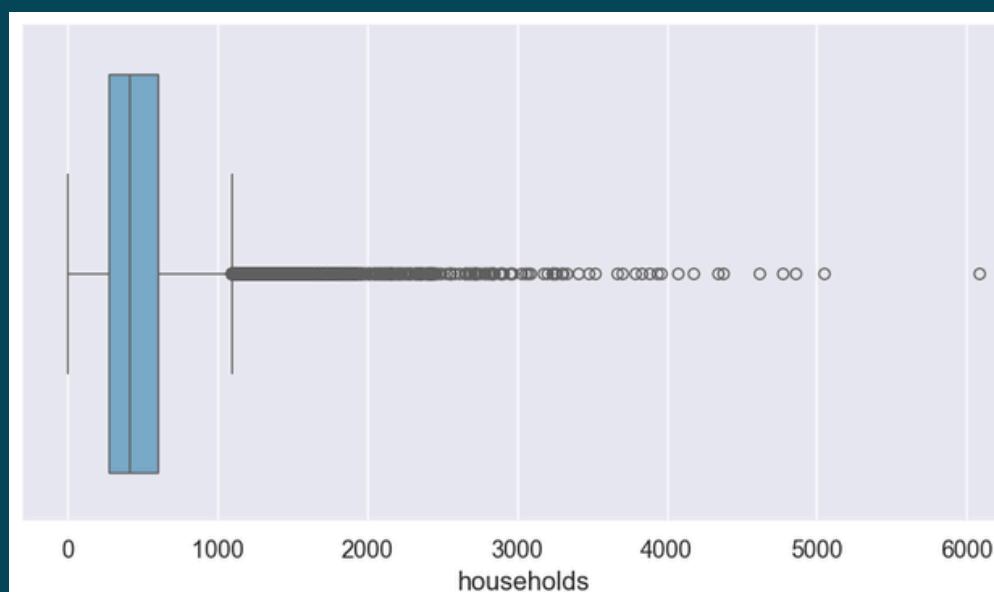
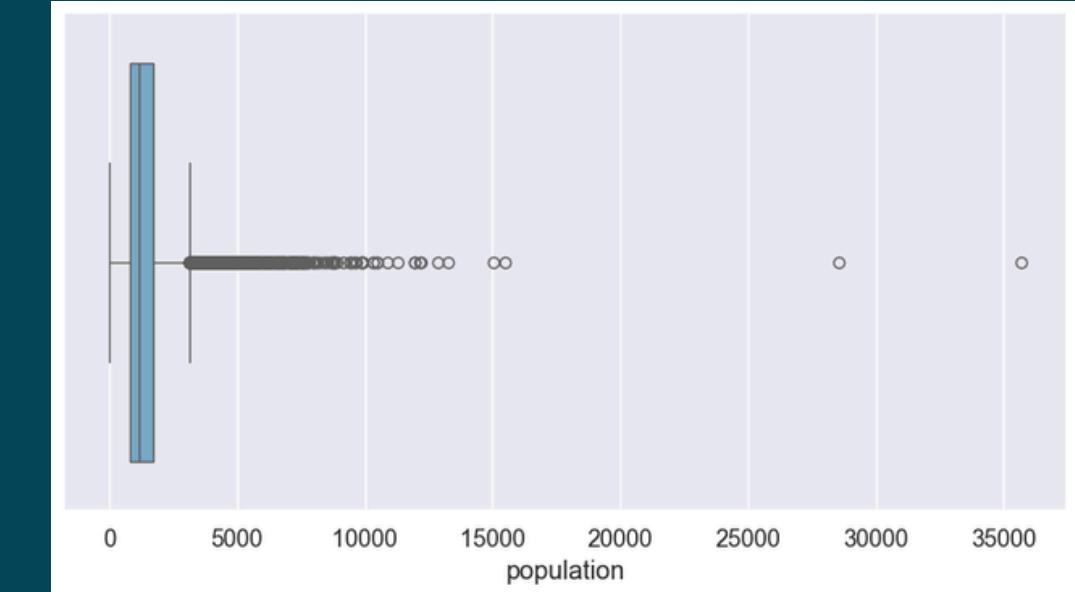
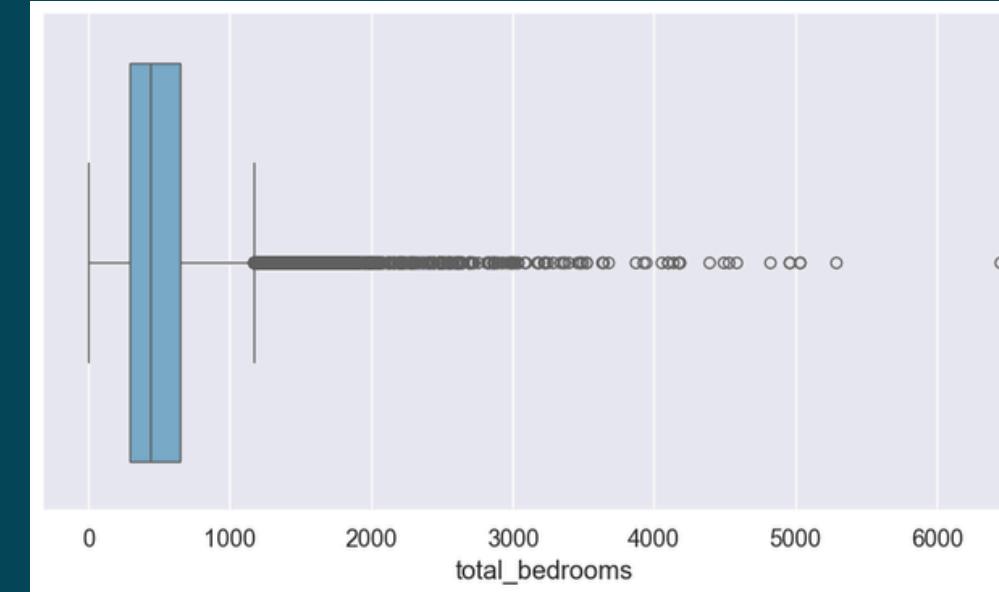
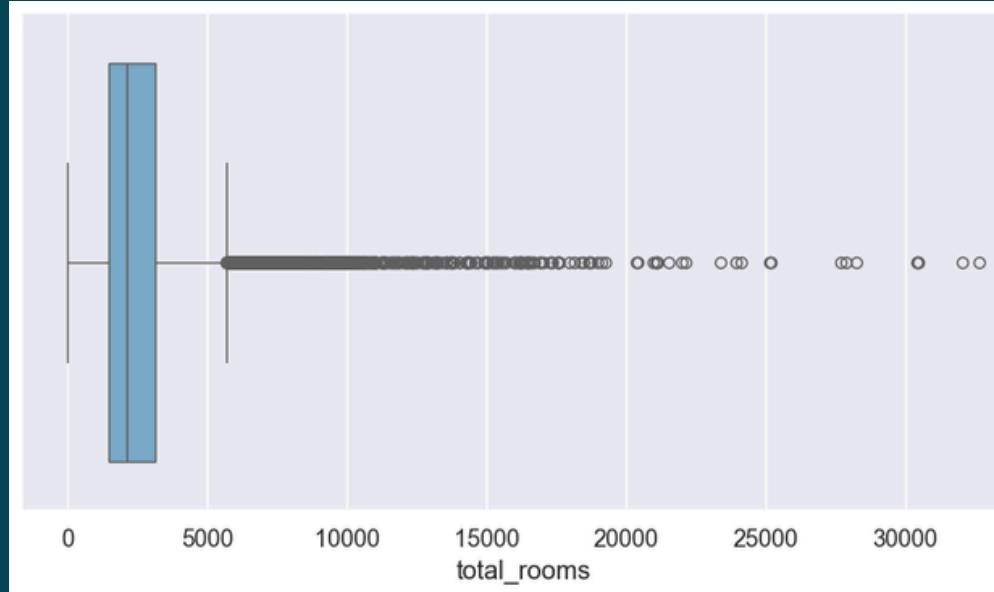
Membuktikan bahwa `ocean_proximity` adalah fitur penting  
(informative feature) dalam prediksi harga rumah.

# EDA



Cocok untuk  
menunjukkan lokasi  
geografis sebagai fitur  
penting dalam model  
prediksi harga rumah.

# Outlier



`df.drop(outlier, inplace=True)` digunakan karena pendekatan tersebut ingin membersihkan data agar model lebih stabil dan akurat, terutama jika memakai model-model seperti linear regression.

# Correlations



Dalam konteks properti atau real estate, memiliki fitur yang sangat mirip (seperti **total\_rooms** dan **total\_bedrooms** atau **households** dan **population**) bisa membingungkan model dalam menentukan harga rumah yang tepat.

Misalnya, dua rumah dengan jumlah kamar yang mirip tetapi populasi yang berbeda bisa memiliki nilai yang sangat berbeda tergantung pada lokasinya.

# Multicollinearity

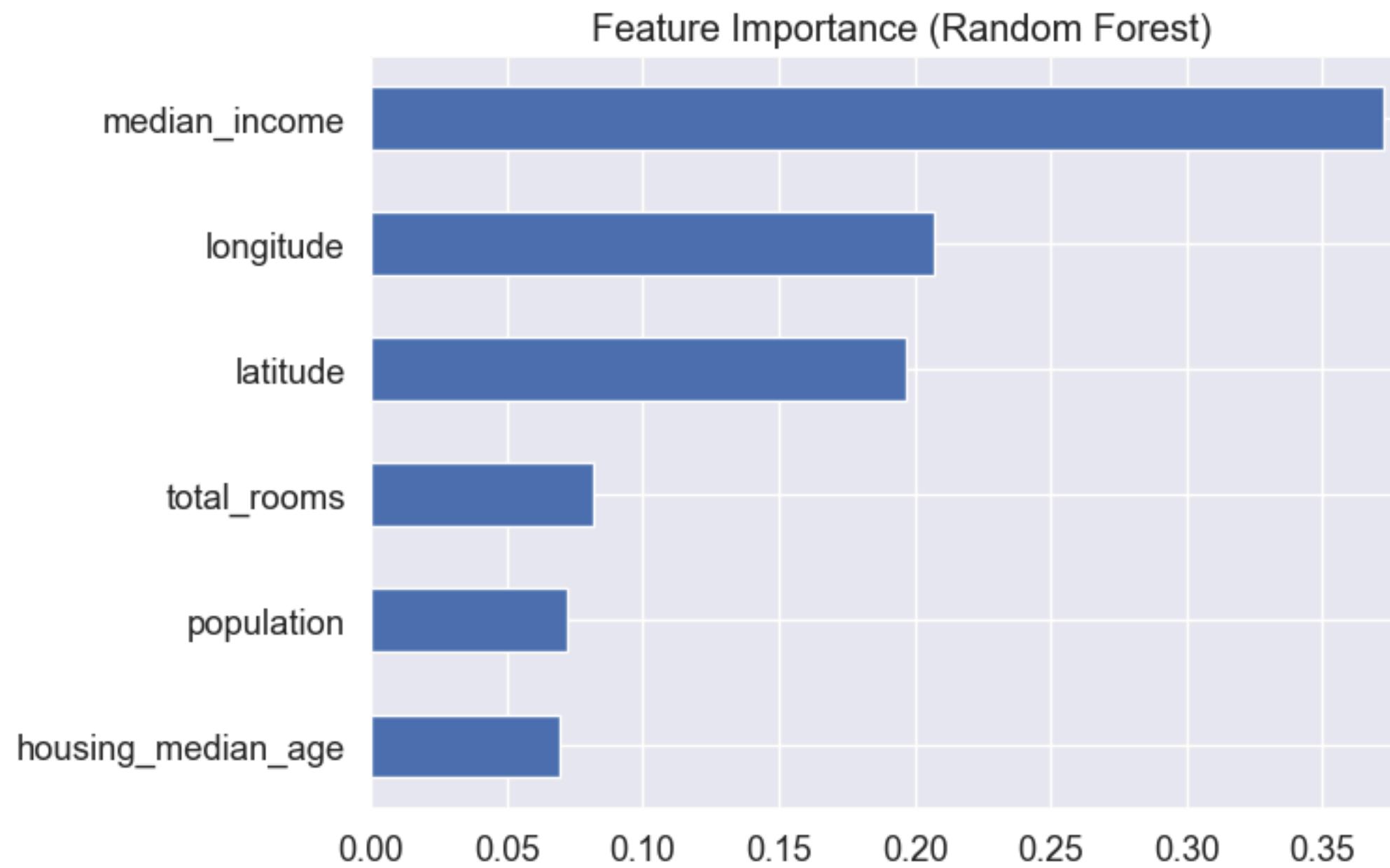
Feature	VIF
const	17223.66
households	21.58
total_bedrooms	19.89
total_rooms	9.64
latitude	9.6
longitude	9.12
population	4.19
median_income	2.14
housing_median_age	1.24

Fitur **households** dan **total\_bedrooms** ini memiliki VIF jauh di atas 5, menandakan korelasi yang sangat tinggi dengan fitur lain.

Ini akan membuat model sulit menentukan hubungan independen antara variabel dan target (median\_house\_value), menyebabkan koefisien yang tidak stabil dan hasil prediksi yang kurang akurat.

Maka dari itu perlu di drop.

# Feature Selection



**median\_income** memiliki pengaruh yang jauh lebih besar dibanding fitur lain, menunjukkan bahwa penghasilan median sangat kuat dalam menentukan nilai rumah, yang masuk akal karena wilayah dengan penghasilan lebih tinggi cenderung memiliki nilai properti yang lebih tinggi.

# Column Encode

## One-hot Encoding

```
df_encoded = pd.get_dummies(df, columns=['ocean_proximity'], drop_first=True)
```

```
df_encoded = df_encoded.select_dtypes(include=[np.number])
```

```
df_encoded = df_encoded.replace([np.inf, -np.inf], np.nan).dropna()
```

# Data Splitting

```
X =  
df_encoded.drop("median_house_valu  
e", axis=1)
```

---

```
y =  
df_encoded["median_house_value"]
```

---

```
x_train, x_test, y_train, y_test =  
train_test_split(X, y, test_size=0.2,  
random_state=123)
```

# Feature Scaling

```
scaler = StandardScaler()  
→ Membuat objek  
StandardScaler
```

---

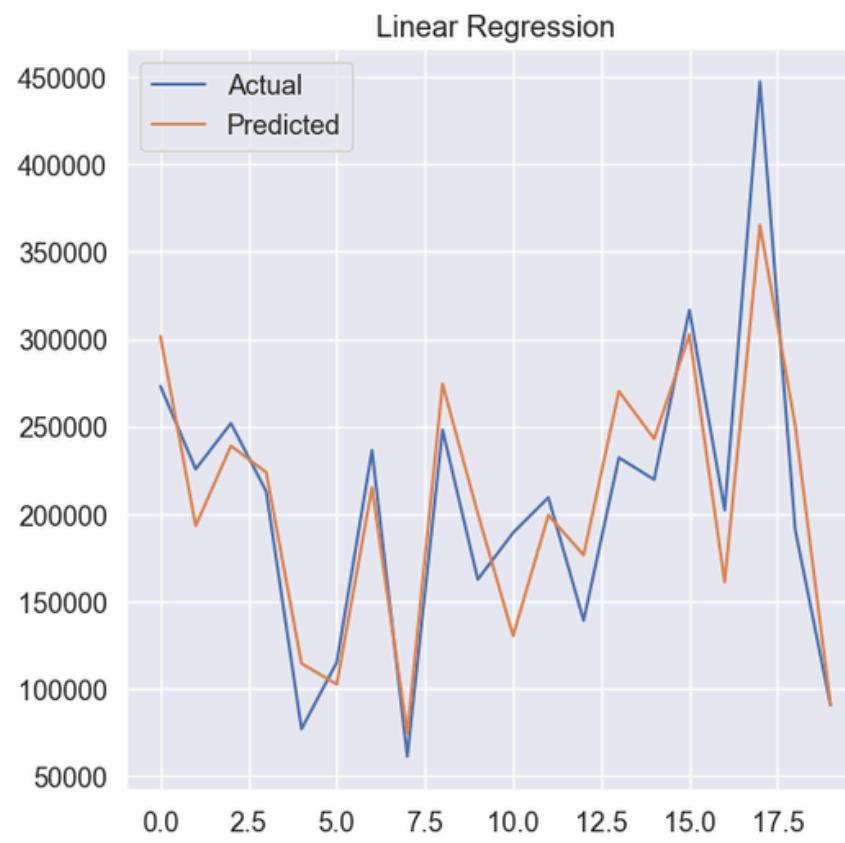
```
x_train_scaled =  
scaler.fit_transform(x_train)
```

---

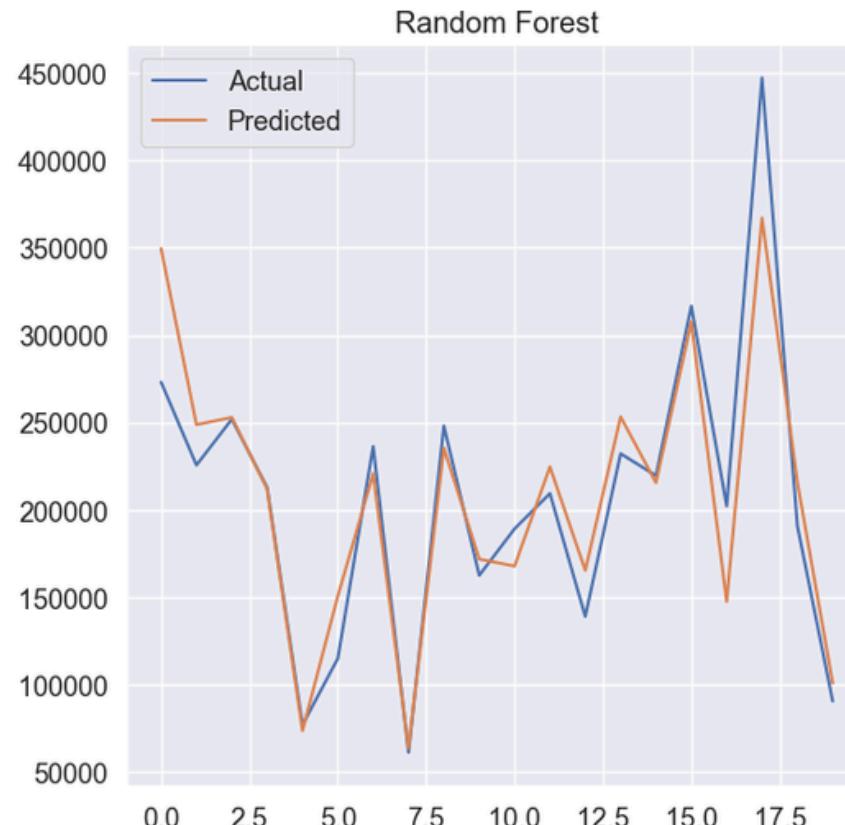
```
x_test_scaled =  
scaler.transform(x_test)
```

# Modelling

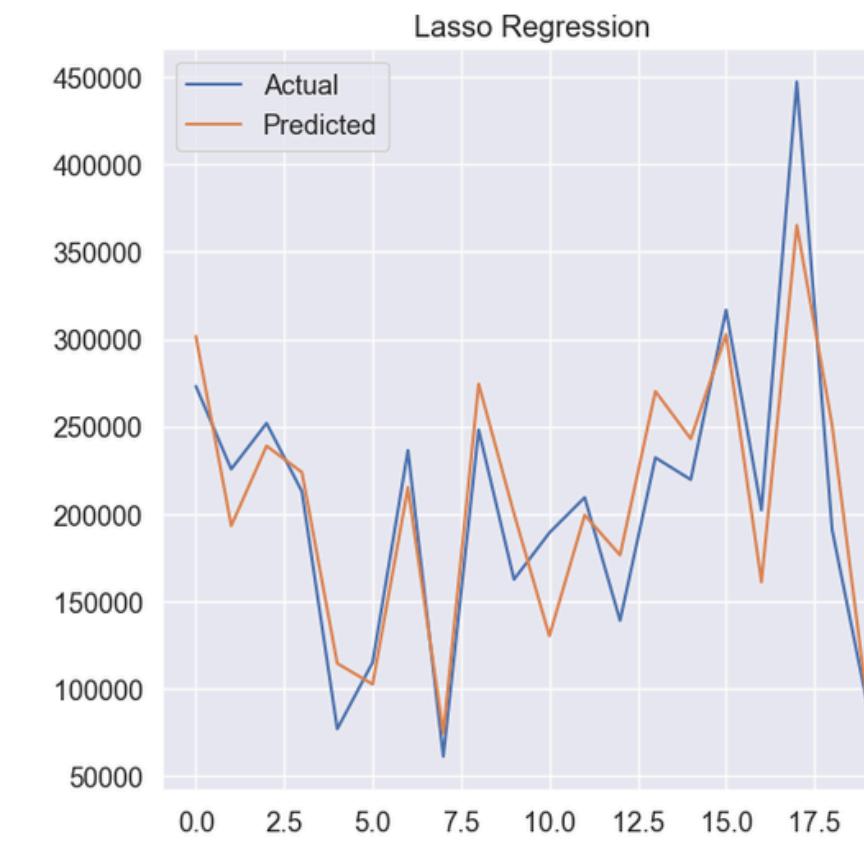
Linier  
Regression



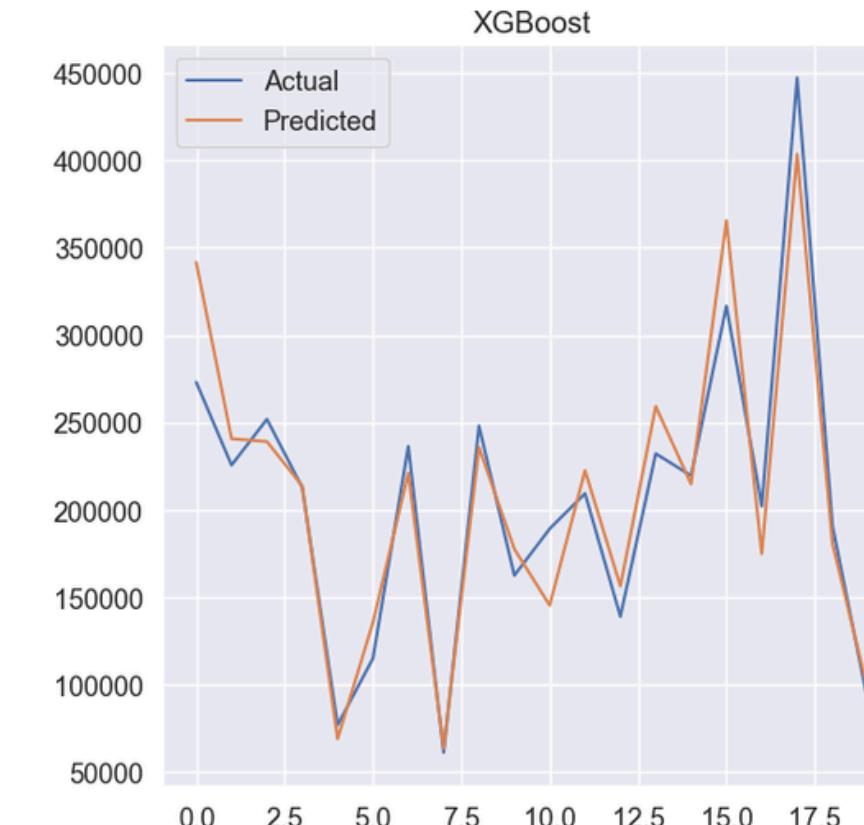
Random Forest  
Regressor



Lasso/Ridge  
Regressors



XGBoost  
Regressor



# Data Insight

Berikut ini adalah interpretasi dari grafik training dan test setiap model

## Linear Regression / Lasso / Ridge

- \* Garis prediksi (orange) cukup sejajar dengan actual (biru), tetapi:
- \* Pola fluktuasi ekstrem (misal naik-turun tajam di data ke-5 dan ke-17) tidak tertangkap dengan baik.
- \* Error masih terlihat besar di beberapa titik.

## Random Forest

- \* Lebih baik menangkap pola fluktuasi tajam, misalnya pada titik 5–6 dan 15–17.
- \* Tapi ada beberapa overfitting kecil di awal data (titik 0–3), di mana prediksi agak terlalu tinggi.

## XGBoost

- \* Menangkap pola lebih smooth dan konsisten, bahkan di titik ekstrem.
- \* Mampu mengikuti arah tren lebih akurat daripada model linear.
- \* Tidak terlalu overfit seperti Random Forest → good generalization.

# Evaluation

Model	R <sup>2</sup>	Adj. R <sup>2</sup>	MAE	MSE	RMSE	MAPE	MSPE	MSLE
Linear Regression	0.5531	0.5524	47,436.82	3,970,304,607	63,010.35	0.3076	0.1958	-
Lasso Regression	0.5531	0.5524	47,435.94	3,970,266,033	63,010.05	0.3076	0.1958	-
Ridge Regression	0.5531	0.5524	47,436.40	3,970,286,469	63,010.21	0.3076	0.1958	-
Random Forest	0.7561	0.7557	31,346.30	2,167,105,895	46,552.18	0.1826	0.0941	0.0571
XGBoost	0.7755	0.7752	30,017.14	1,994,174,263	44,656.18	0.176	0.0898	0.0573

# Conclusion

Berdasarkan hasil evaluasi terhadap beberapa model regresi (Linear, Lasso, Ridge, Random Forest, dan XGBoost), dapat disimpulkan bahwa XGBoost Regressor merupakan model dengan performa terbaik. Hal ini didasarkan pada nilai evaluasi sebagai berikut:

- R-squared ( $R^2$ ): 0.775 → tertinggi di antara semua model, menandakan bahwa model mampu menjelaskan variabilitas data dengan sangat baik.
- MAE (Mean Absolute Error): 30,017 → nilai kesalahan rata-rata terkecil.
- RMSE (Root Mean Square Error): 44,656 → paling kecil, menunjukkan akurasi prediksi yang tinggi.
- RMSE (Root Mean Square Error): 44,656 → paling kecil, menunjukkan akurasi prediksi yang tinggi.
- MAPE (Mean Absolute Percentage Error): 17.6% → error relatif paling kecil, artinya prediksi relatif dekat dengan harga aktual.
- MSPE & MSLE: nilai terbaik di antara model lain.

# Recommendation

- 1 GUNAKAN MODEL XGBOOST REGRESSOR**
- 2 TENTUKAN KISARAN HARGA YANG DIREKOMENDASIKAN**
- 3 TAMBAHKAN FITUR RELEVAN**
- 4 PERLUAS DATASET**
- 5 PERTIMBANGKAN PEMBARUAN MODEL SECARA BERKALA**

Thank  
You