

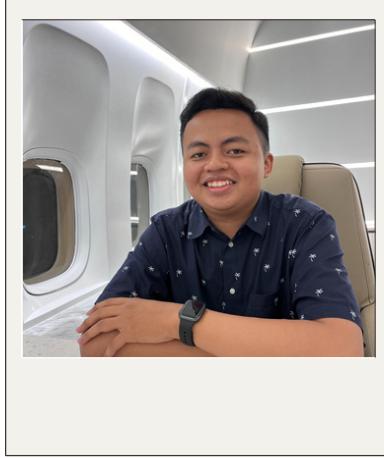


House Price Prediction

Margareth Hamilton

Friday, November 8th 2024

Meet the Team



Alwy Bathia R.



Daniel Machsimus L.



Jason Hermawan

Background & Problem Statement

Kebutuhan akan adanya **prediksi harga rumah** terus bermunculan dengan **pemenuhan kebutuhan primer manusia**. Tidak hanya diaplikasikan pada pembeli rumah, prediksi harga rumah dapat dimanfaatkan penjual (realtor), agen properti, dan investor dalam **penetapan harga jual rumah** yang **kompetitif** [1].

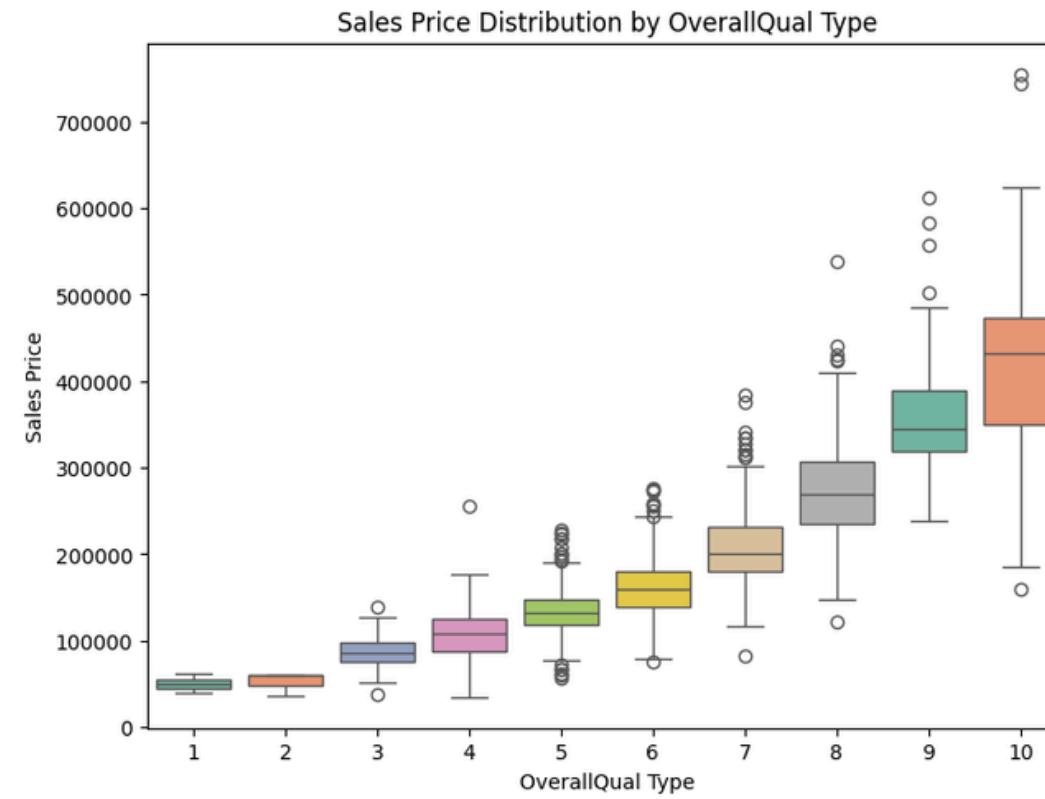
Machine Learning dapat digunakan sebagai *tools* untuk membantu dalam hal *decision-assisting* bagi banyak *stakeholder* [2].

Dataset yang digunakan adalah **Ames Housing Dataset**. Dataset ini berisi berbagai **feature** terkait perumahan di daerah Ames, Iowa, US. Dataset ini memang sering digunakan untuk prediksi harga rumah.

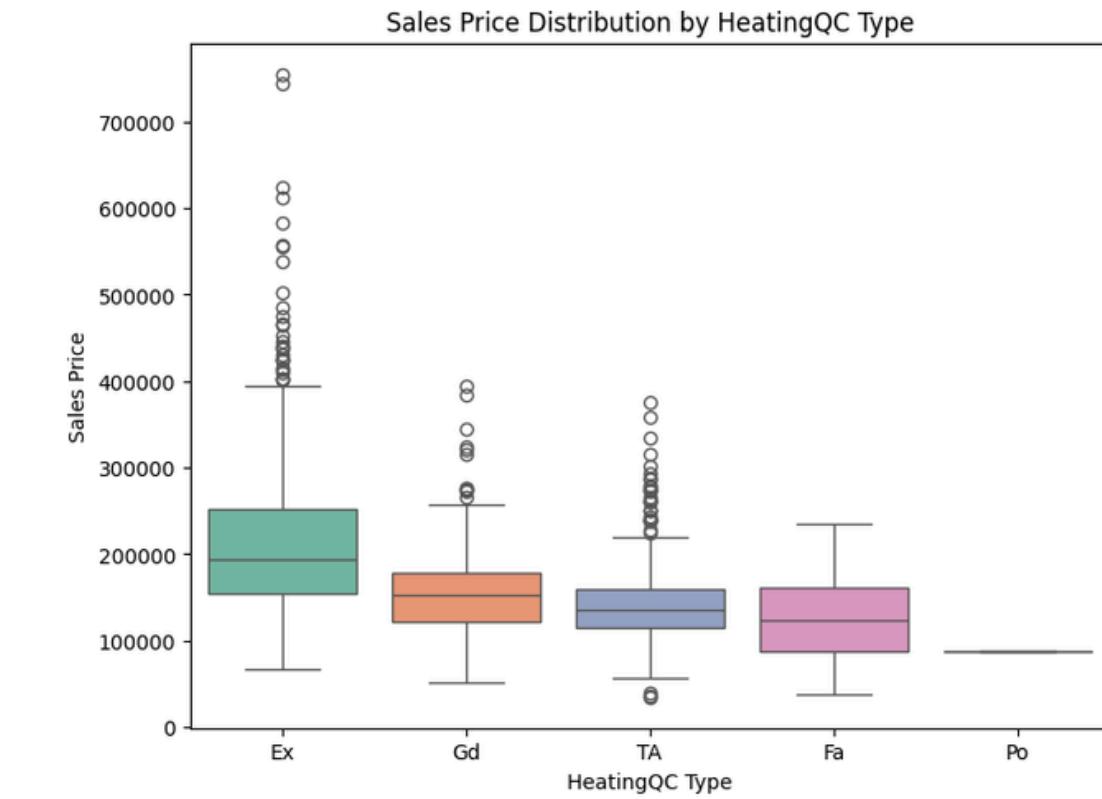


Exploratory Data Analysis Explore Variables

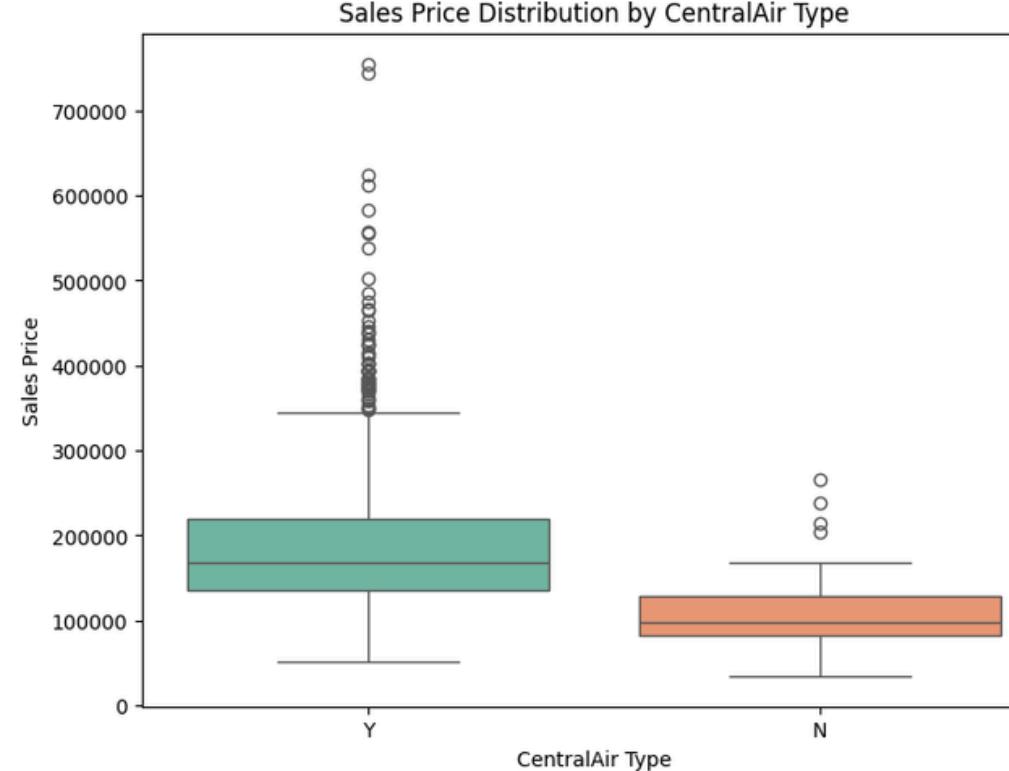
Terdapat 3 grafik yang menurut kami dilihat dari boxplotnya memiliki perbedaan antarkelas yang berbeda



Terdapat **26** dari total
81 feature yang
memiliki outlier.



Terdapat **26** dari total
81 feature yang
memiliki outlier.



Exploratory Data Analysis NA value

Percentase untuk Data yang terdeteksi ada nilai NA/None-nya

	0	1
LotFrontage	259	17.739726
Alley	1369	93.767123
MasVnrType	872	59.726027
MasVnrArea	8	0.547945
BsmtQual	37	2.534247
BsmtCond	37	2.534247
BsmtExposure	38	2.602740
BsmtFinType1	37	2.534247
BsmtFinType2	38	2.602740
Electrical	1	0.068493
FireplaceQu	690	47.260274
GarageType	81	5.547945
GarageYrBlt	81	5.547945
GarageFinish	81	5.547945
GarageQual	81	5.547945
GarageCond	81	5.547945
PoolQC	1453	99.520548
Fence	1179	80.753425
MiscFeature	1406	96.301370

Terdapat **19** dari total **81 feature** yang memiliki **nilai NA/null**.

Data yang memang variasi nilainya ada NA-nya:

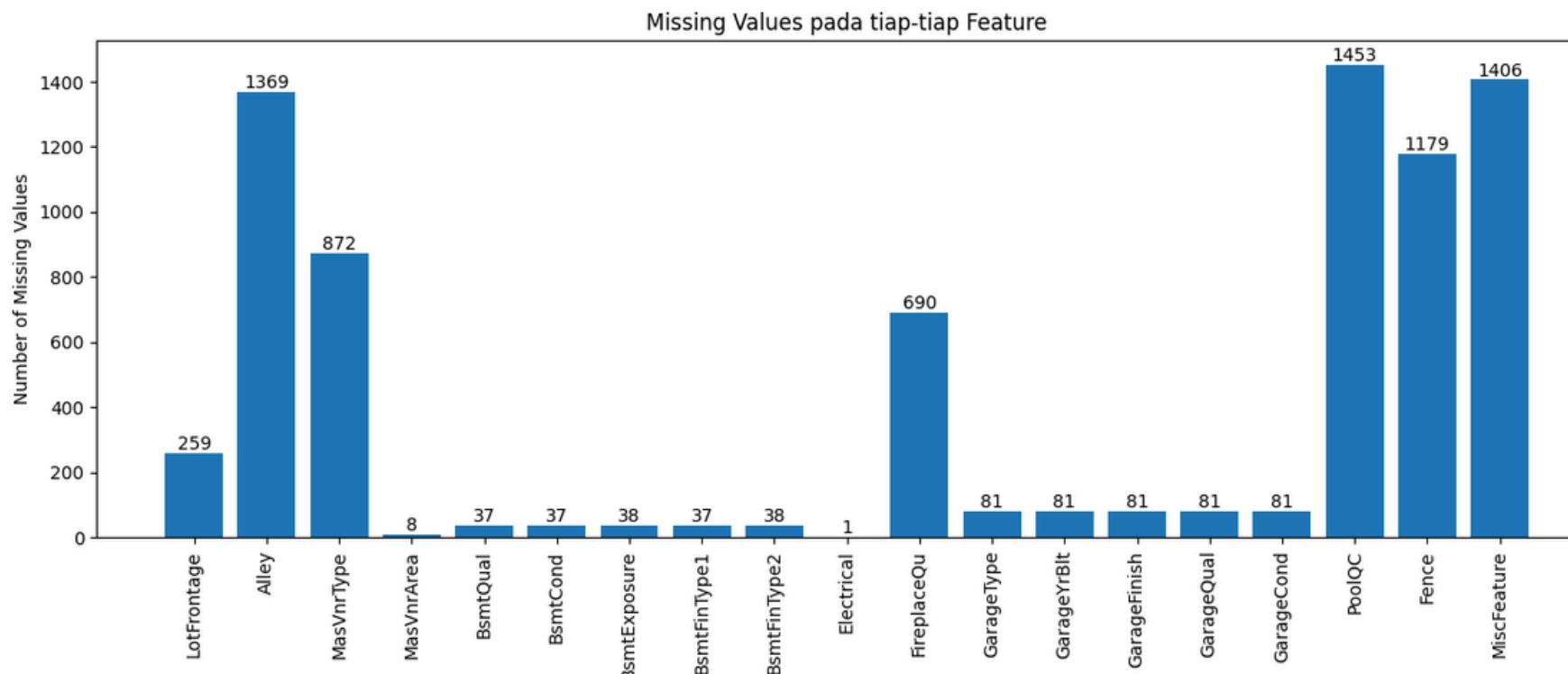
- Alley (No Alley Access)
- PoolQC (No Pool)
- MasVnrType (None)
- BsmtQual (No basement)
- BsmtCond (No basement)
- BsmtExposure (No basement)
- BsmtFinType1 (No basement)
- BsmtFinType2 (No basement)
- FireplaceQu (No Fireplace)
- GarageType (No Garage)
- GarageFinish (No Garage)
- GarageQual (No Garage)
- GarageCond (No Garage)
- MiscFeature (None)
- Fence (No Fence)

Memang ada **15 feature** kategorikal memiliki variasi nilainya adalah nilai **NA/None**.

Untuk itu, ada **perlakuan** yang berbeda:

(1) perlakuan untuk **15 feature** yang memang bervariasi “NA” dan

(2) perlakuan untuk **4 feature** yang benar-benar hilang



Exploratory Data Analysis NA value

Untuk itu, ada **perlakuan** yang berbeda:

(1) perlakuan untuk **15 feature** yang memang bervariasi "NA" -> **imputasi dengan nilai spesifik**

Data yang memang variasi nilainya ada NA-nya:

- Alley (No Alley Access)
- PoolQC (No Pool)
- MasVnrType (None)
- BsmtQual (No basement)
- BsmtCond (No basement)
- BsmtExposure (No basement)
- BsmtFinType1 (No basement)
- BsmtFinType2 (No basement)
- FireplaceQu (No Fireplace)
- GarageType (No Garage)
- GarageFinish (No Garage)
- GarageQual (No Garage)
- GarageCond (No Garage)
- MiscFeature (None)
- Fence (No Fence)

Mengganti **nilai NA/None** menjadi variasi yang **sesuai** dengan **variasi None masing-masing feature**

Untuk itu, ada **perlakuan** yang berbeda:

(2) perlakuan untuk **4 feature** yang benar-benar hilang -> **imputasi**

Percentase untuk Data yang terdeteksi ada nilai NA/None-nya

	0	1	
LotFrontage	259	17.739726	Numerikal
MasVnrArea	8	0.547945	Numerikal
Electrical	1	0.068493	Kategorikal
GarageYrBlt	81	5.547945	Numerikal

→ Mean
→ nilai 0 (nol)
→ Modus
→ nilai 0 (nol) → Feature Engineering

LotFrontage

Mean

GarageYrBlt

GarageYrBlt yang NA artinya rumah tidak memiliki garasi.

MasVnrArea

MasVnrArea yang NA artinya rumah tidak memiliki MasVnr

Electrical

Data kategorikal -> modus paling umum

Modelling

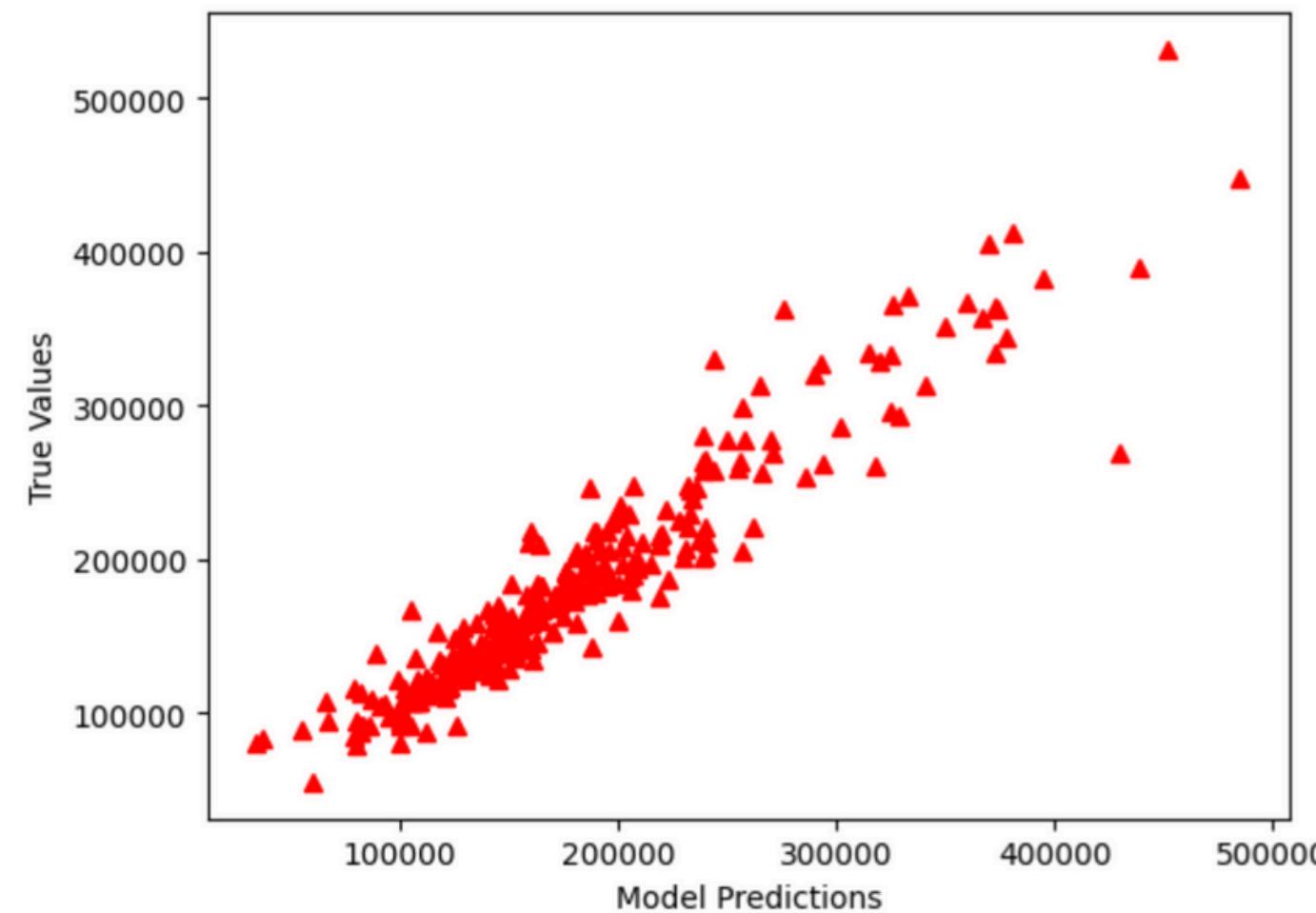
Pada Project ini kami menggunakan 2 Model

1. XGBOOST REGRESSOR

2. SUPPORT VECTOR REGRESSION

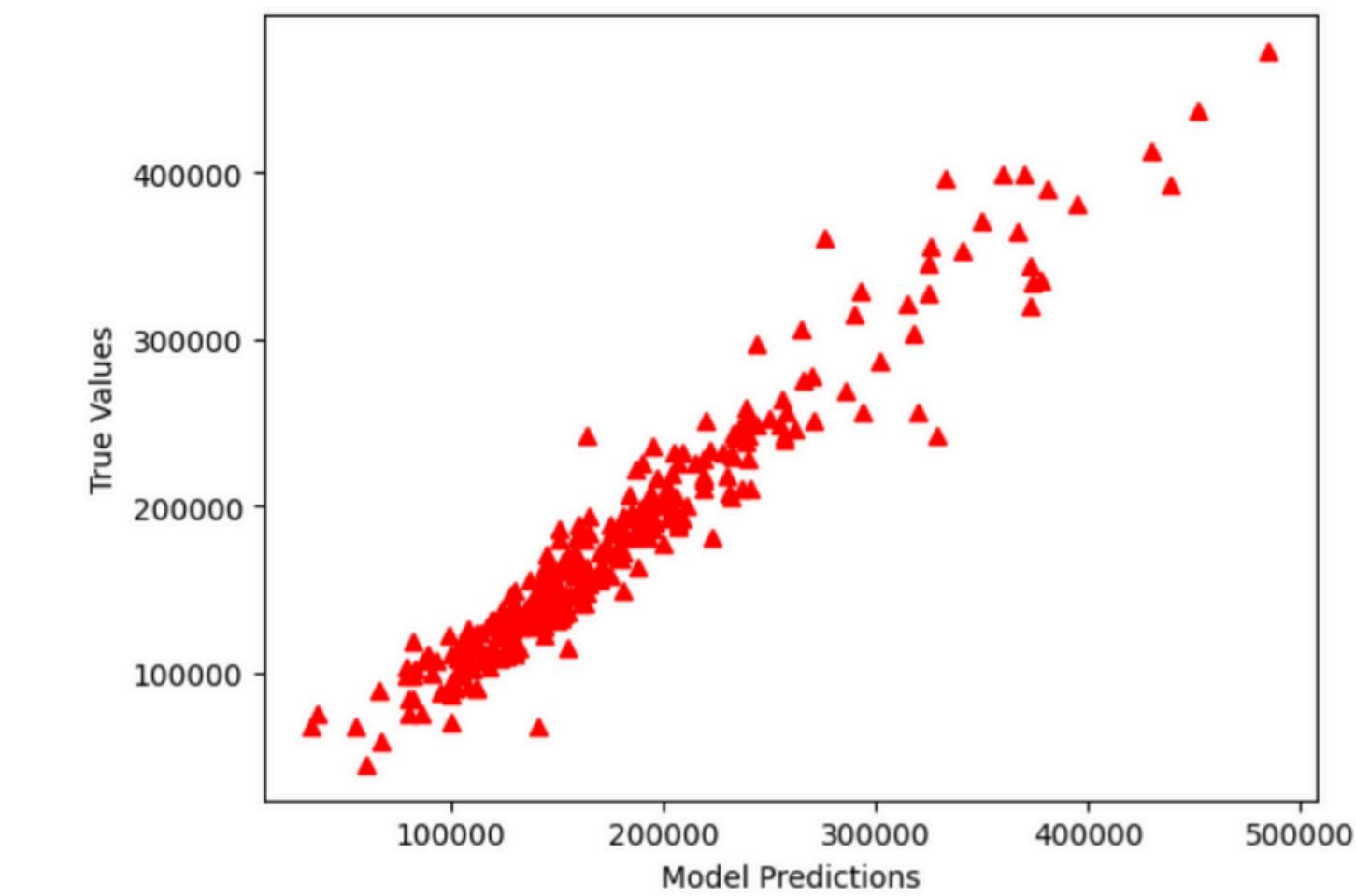
Modelling XGBOOST

BEFORE GRIDSEARCH



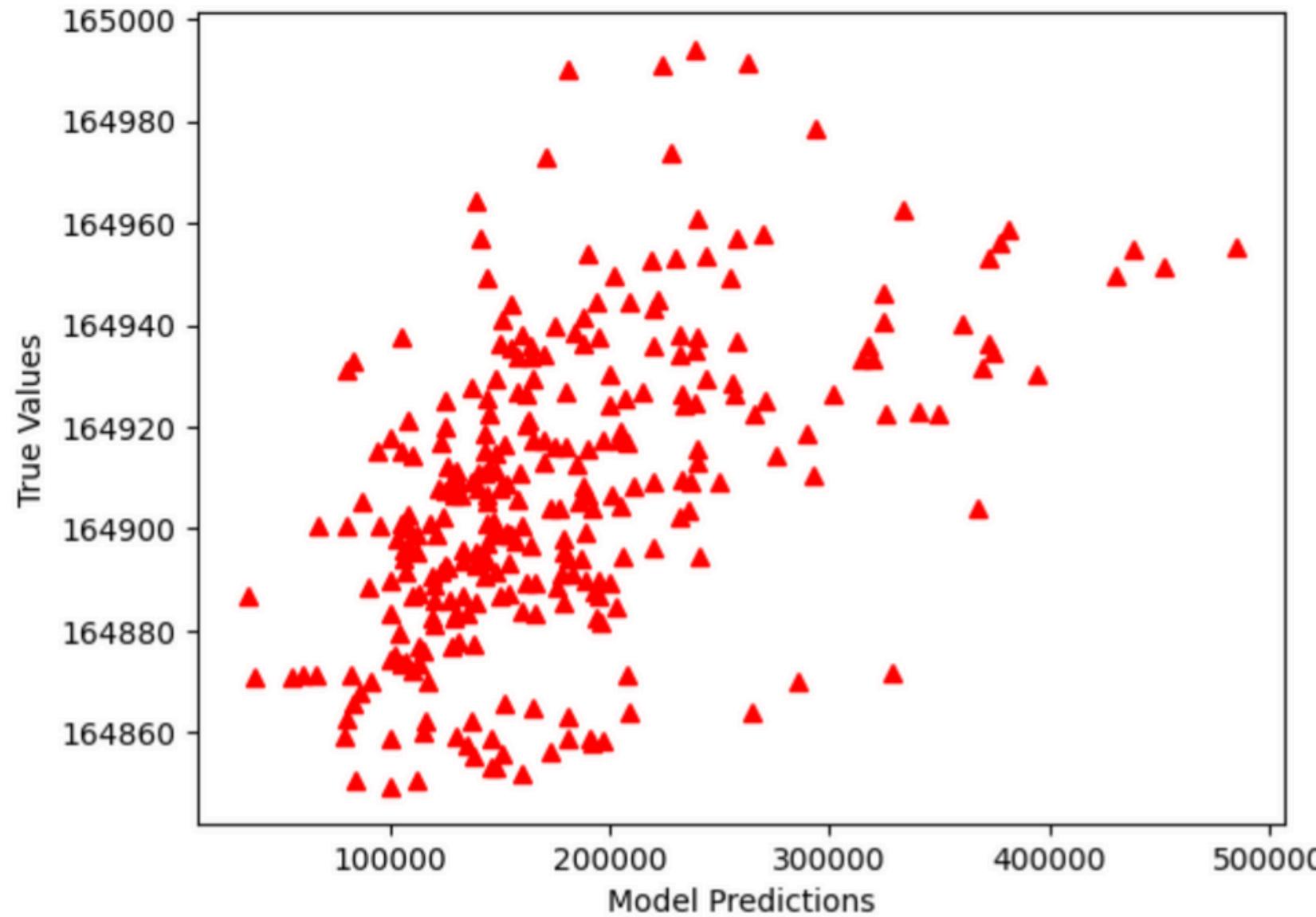
RMSE = 23574.021
MSE = 555734500.0
MAE = 15924.144
R2 = 0.9029815196990967
Adjusted R2 = 3.5665797970511695

AFTER GRIDSEARCH



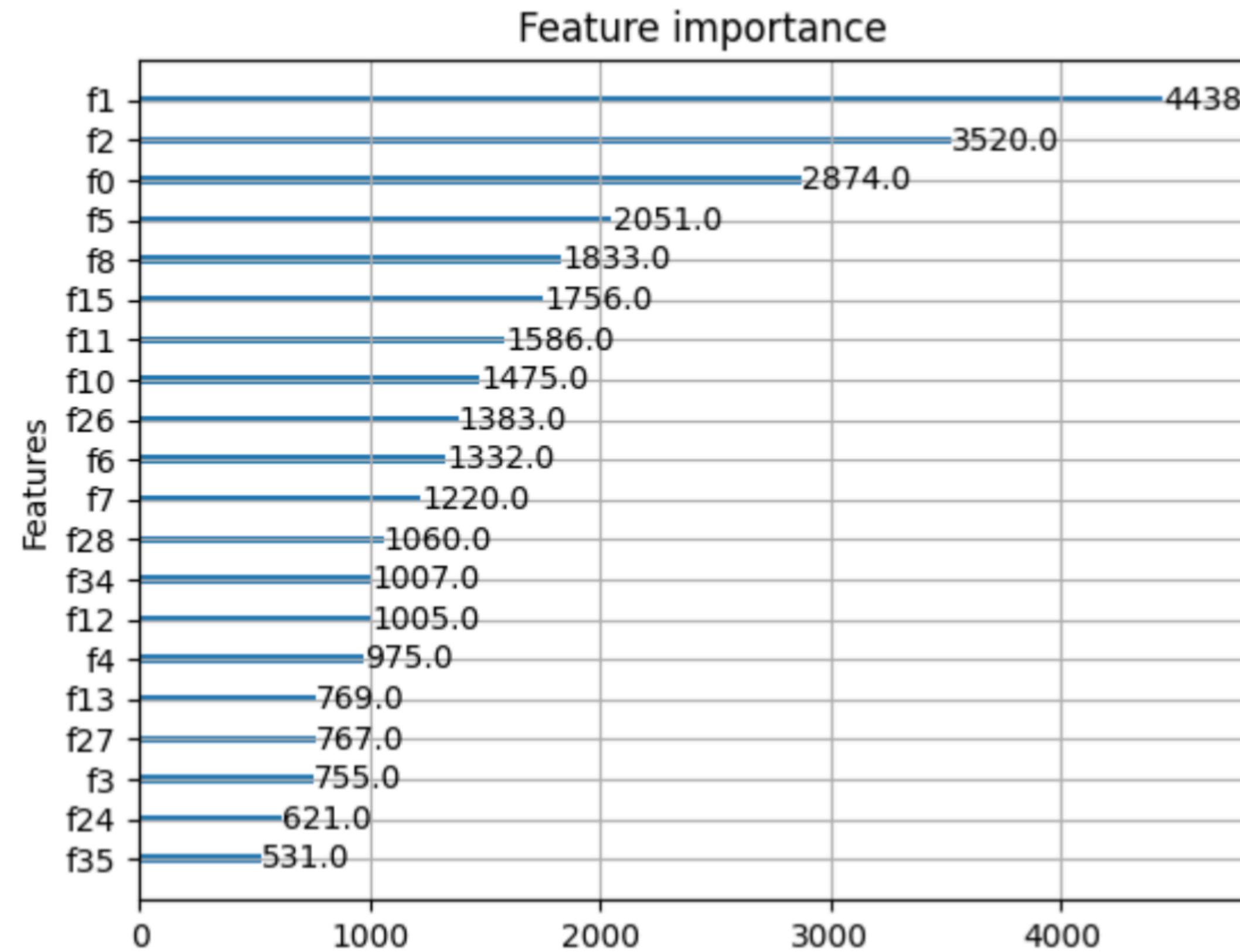
RMSE = 19438.764
MSE = 377865570.0
MAE = 13934.45
R2 = 0.934033342552185
Adjusted R2 = 2.7451181574301287

MODELLING SVR



RMSE = 76681.091
MSE = 5879989724.456878
MAE = 55048.02728110719
R2 = -0.026511244663566913
Adjusted R2 = 28.155888381554362

SUMMARY



F0 :MSSubClass
F1:LotFrontage
F2: LotArea
F5:YearBuilt
F8: BsmtFinSF1

SUMMARY

- Perbandingan dari ke 3 Model ini terlihat cukup signifikan, XGBOOST sebelum menggunakan GRIDSEARCH memiliki performa yang lebih baik.
- Meskipun komputasi GRIDSEARCH cukup lama namun metode ini sangat membantu untuk menemukan parameter terbaik dan meningkatkan akurasi pada model XGBOOST
- Dari hasil feature Importance terlihat bahwa model dapat mengidentifikasi faktor apa saja yang berpengaruh terhadap harga jual rumah dan hasil ini sejalan dengan keadaan bisnis yang ada.

Model Comparison

Pada bagian ini, kami mencoba membandingkan beberapa model untuk mencari model yang terbaik.

Berikut beberapa model yang kami Gunakan:

1. Linear Regression
2. XGBoost (tanpa hypertuning)
3. XGBoost (dengan GridSearch hypertuning)

Linear Regression

```
[18]: # Evaluate the model
mse = mean_squared_error(y_test, y_pred_linear)
mae = mean_absolute_error(y_test, y_pred_linear)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred_linear)
mape = mean_absolute_percentage_error(y_test, y_pred_linear) * 100

[19]: print(f"Mean Absolute Error (MAE): {mae}")
print(f"Mean Squared Error (MSE): {mse}")
print(f"Root Mean Squared Error (RMSE): {rmse}")
print(f"R2 Score: {r2}")
print(f"Mean Absolute Percentage Error (MAPE): {mape}%")


Mean Absolute Error (MAE): 21836.050017510923
Mean Squared Error (MSE): 1211435838.551941
Root Mean Squared Error (RMSE): 34805.686870854035
R2 Score: 0.8420620185075283
Mean Absolute Percentage Error (MAPE): 13.06471784770786%
```

XGBoost (Without Hypertuning)

```
[25]: # Evaluate the model
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)
mape = mean_absolute_percentage_error(y_test, y_pred) * 100

[26]: print(f"Mean Absolute Error (MAE): {mae}")
print(f"Mean Squared Error (MSE): {mse}")
print(f"Root Mean Squared Error (RMSE): {rmse}")
print(f"R2 Score: {r2}")
print(f"Mean Absolute Percentage Error (MAPE): {mape}%")


Mean Absolute Error (MAE): 15632.952656785103
Mean Squared Error (MSE): 617930642.6687983
Root Mean Squared Error (RMSE): 24858.210769659152
R2 Score: 0.9194388389587402
Mean Absolute Percentage Error (MAPE): 9.556709724005907%
```

XGBoost (Grid Search Hypertuning)

```
[28]: mse_tuned = mean_squared_error(y_test, y_pred_tuned)
      mae_tuned = mean_absolute_error(y_test, y_pred_tuned)
      rmse_tuned = np.sqrt(mse_tuned)
      r2_tuned = r2_score(y_test, y_pred_tuned)
      mape = mean_absolute_percentage_error(y_test, y_pred_tuned) * 100

[29]: print("\nEvaluation metrics for the tuned model:")
      print(f"Tuned Mean Absolute Error (MAE): {mae_tuned}")
      print(f"Tuned Mean Squared Error (MSE): {mse_tuned}")
      print(f"Tuned Root Mean Squared Error (RMSE): {rmse_tuned}")
      print(f"Tuned R2 Score: {r2_tuned}")
      print(f"Mean Absolute Percentage Error (MAPE): {mape}%")
```

```
Evaluation metrics for the tuned model:
Tuned Mean Absolute Error (MAE): 16799.78712275257
Tuned Mean Squared Error (MSE): 654524968.3826722
Tuned Root Mean Squared Error (RMSE): 25583.685590287263
Tuned R2 Score: 0.9146679043769836
Mean Absolute Percentage Error (MAPE): 10.230484863494707%
```

Model Comparison

Dari berbagai model yang sudah dicoba, hasil terbaik ada di model
XGBoost tanpa GridSearch Hypertuning.

Conclusion

Kesimpulan yang dapat diambil dari projek House Price Prediction ini adalah, **AI dapat sangat membantu dalam melakukan prediksi harga rumah di masa depan.**

Sehingga kita dapat membuat keputusan-keputusan berdasarkan prediksi yang telah dilakukan oleh model AI ini.

THANK YOU