

Comparative Analysis of K-Nearest Neighbors Algorithm and Random Forest Regressor for House Price Prediction in Bandung City

Dimas Yudhistira Ananda¹, Ferian Fauzi Abdulloh²

Informatika, Ilmu Komputer, Universitas Amikom Yogyakarta, Indonesia

dimasyudhistira29@students.amikom.ac.id¹, ferian@amikom.ac.id²

Article Info

Article history:

Received 2025-08-08

Revised 2025-11-15

Accepted 2026-02-10

Keyword:

KNN Regressor,
Random Forest Regressor,
House Price Prediction,
Machine Learning,
Real Estate Valution,
Bandung

ABSTRACT

The rapid population growth and continuous urban expansion in Bandung have contributed to volatile and escalating housing prices, creating significant challenges for market transparency and affordability. This study aims to develop and evaluate machine-learning models to predict house prices in the Bandung region using a publicly available dataset consisting of 7,609 property records. Following the CRISP-DM methodology, the research includes data exploration, preprocessing (outlier handling using IQR, one-hot encoding, and feature standardization), model training, and performance evaluation. Two regression models K-Nearest Neighbors (KNN) Regressor and Random Forest (RF) Regressor—were compared through systematic hyperparameter tuning using Grid Search and Random Search techniques. The experimental results show that the Random Forest Regressor achieves the best performance with an R² score of 0.7838 and a mean absolute error (MAE) of approximately Rp 399.7 million, outperforming the optimized KNN model. Feature importance analysis also indicates that land area, building area, and location are the most influential predictors of property prices. The findings highlight the effectiveness of ensemble methods in handling complex real-estate data and demonstrate the potential of machine-learning-based predictive tools to support buyers, sellers, and policymakers in making informed and data-driven decisions in the Bandung housing market.



This is an open access article under the [CC-BY-SA](#) license.

I. PENDAHULUAN

Pertumbuhan populasi dan laju urbanisasi yang pesat merupakan fenomena kunci yang membentuk lanskap sosial-ekonomi di Indonesia. Provinsi Jawa Barat, sebagai provinsi dengan jumlah penduduk terpadat, mengalami tekanan demografis yang signifikan, terutama di pusat-pusat urban seperti Kota Bandung dan sekitarnya [1]. Sebagai ibu kota provinsi yang juga menjadi pusat pariwisata, pendidikan, dan industri kreatif, Tingkat pertumbuhan kota Bandung yang mendorong kondisi investasi yang menguntungkan menjadikan kota ini sebagai wilayah yang menjanjikan dalam sektor perumahan [2]. Penelitian sebelumnya menunjukkan bahwa harga tanah di beberapa wilayah Bandung telah mencapai kisaran Rp3,4 juta hingga Rp5 juta per meter persegi pada tahun 2019 (Somantri, 2020). Temuan historis ini menggambarkan bahwa tren kenaikan harga tanah di

Bandung telah terjadi sejak beberapa tahun sebelumnya dan terus berlanjut hingga saat ini [3]. Tidak hanya meningkat, pasar properti di Bandung juga mengalami fluktuasi harga yang cukup signifikan dari tahun ke tahun.

Ketidakstabilan harga ini dipicu oleh dinamika permintaan, kebijakan makroekonomi, dan perkembangan wilayah yang tidak merata. Studi menunjukkan bahwa *“perluasan yang sangat cepat telah membebani ketersediaan lahan, meningkatkan harga properti, dan membatasi area pengembangan strategis”* (Safitri et al., 2025). Selain itu, fenomena urban sprawl juga berdampak langsung pada keterjangkauan hunian karena *“keputusan penggunaan lahan sangat dipengaruhi oleh pertimbangan finansial yang kemudian menentukan tingkat pelebaran kota”* (Safitri et al., 2025). Dampak ekonomi dan sosial dari ketidakstabilan harga rumah ini sangat terasa bagi masyarakat. Lonjakan harga yang tidak terprediksi menyebabkan masyarakat berpenghasilan

rendah dan menengah semakin sulit membeli rumah, meningkatkan risiko perpindahan ke wilayah pinggiran, dan memperluas kesenjangan sosial. Dari sisi kebijakan perumahan, ketidakstabilan harga mempersulit pemerintah daerah dalam merancang program subsidi, menentukan zonasi, serta menetapkan kebijakan tata ruang yang efektif. Fenomena ini sejalan dengan penelitian sebelumnya yang menunjukkan bahwa harga properti di Indonesia mengalami variasi harga dan indeks yang tidak stabil [4].

Kepadatan penduduk yang terus meningkat ini secara langsung mendorong peningkatan permintaan terhadap hunian yang layak. Akibatnya, pemenuhan kebutuhan akan tempat tinggal menjadi prioritas utama bagi berbagai lapisan masyarakat, mulai dari generasi milenial yang pertama kali memasuki pasar *properti* hingga keluarga yang telah mapan, yang tercermin dari tingginya minat masyarakat untuk berinvestasi dalam properti sejak dulu [5].

Seiring dengan tingginya permintaan, pasar properti di Bandung menjelma menjadi sebuah ekosistem yang sangat dinamis dan kompleks. Harga sebuah properti tidak terbentuk secara tungsang, melainkan merupakan hasil interaksi dari berbagai faktor multifaset. Faktor-faktor ini mencakup variabel makroekonomi, kebijakan suku bunga, aksesibilitas lokasi, ketersediaan fasilitas publik, hingga atribut spesifik dari properti itu sendiri seperti luas tanah, luas bangunan, dan jumlah kamar [6]. Kompleksitas ini sering kali menciptakan tantangan signifikan, baik bagi penjual yang kesulitan menetapkan harga yang kompetitif maupun bagi pembeli yang menghadapi ketidakpastian nilai wajar aset. Kondisi ini dapat mengarah pada asimetri informasi, di mana salah satu pihak memiliki pemahaman yang lebih baik tentang pasar, sehingga merugikan pihak lainnya [1].

Menjawab tantangan tersebut, pengembangan model prediksi harga yang akurat dan objektif menjadi sebuah kebutuhan yang mendesak. Metode penilaian konvensional sering kali bersifat subjektif dan memakan waktu. Sebaliknya, kemajuan dalam bidang ilmu data dan kecerdasan buatan menawarkan solusi melalui pendekatan *machine learning* (pembelajaran mesin). Dengan kemampuannya untuk memproses data dalam jumlah besar dan mengenali pola tersembunyi, *machine learning* dapat membangun model prediksi yang tidak hanya cepat tetapi juga berbasis data, sehingga mengurangi bias dan meningkatkan akurasi [7]. Pendekatan ini berperan penting dalam analisis pasar properti, membantu individu maupun perusahaan untuk membuat keputusan investasi yang lebih baik.

Untuk mengimplementasikan pendekatan tersebut, penelitian ini secara spesifik bertujuan untuk mengembangkan dan membandingkan kinerja dua algoritma *machine learning* dengan metodologi yang kontras: K-Nearest Neighbors (KNN) Regressor dan Random Forest (RF) Regressor. Model pertama, KNN Regressor, dipilih karena merupakan algoritma berbasis instans yang mampu menangkap pola non-linear secara intuitif tanpa memerlukan proses pelatihan yang kompleks. Algoritma ini memprediksi

harga rumah berdasarkan rata-rata nilai dari ‘k’ tetangga terdekatnya.

Namun, kinerja KNN sangat sensitif terhadap pencilan (*outliers*) dan skala data, sehingga memerlukan normalisasi fitur untuk mencapai hasil yang optimal [8]. Sebagai bandingan, dipilih Random Forest Regressor, sebuah algoritma *ensemble* yang memanfaatkan kekuatan agregat dari banyak *decision trees* (pohon keputusan). Dengan menggunakan pendekatan *bagging*, Random Forest membangun model yang lebih stabil dan tahan terhadap *overfitting*. Salah satu keunggulan utamanya adalah kemampuannya untuk mengukur dan mengidentifikasi fitur-fitur yang paling berpengaruh dalam menentukan harga rumah, memberikan wawasan yang lebih dalam selain dari sekadar prediksi [9].

Dengan membandingkan kedua model ini—satu berbasis kedekatan (KNN) dan yang lainnya berbasis ansambel pohon keputusan (RF)—penelitian ini bertujuan untuk mengidentifikasi pendekatan mana yang memberikan tingkat akurasi dan keandalan tertinggi untuk pasar properti Bandung. Kontribusi utama dari penelitian ini adalah menyediakan sebuah model prediktif yang tervalidasi serta analisis komparatif yang dapat dimanfaatkan oleh calon pembeli, penjual, agen properti, dan investor sebagai referensi objektif dalam proses pengambilan keputusan strategis.

II. TINJAUAN PUSTAKA

Prediksi harga rumah menggunakan pendekatan *machine learning* telah menjadi fokus banyak penelitian. Berbagai algoritma telah diterapkan untuk memodelkan hubungan kompleks antara fitur properti dan harganya. Tinjauan ini akan membahas beberapa studi relevan yang berfokus pada algoritma yang digunakan dalam penelitian ini dan bandingnya.

2.1 Studi Mengenai K-Nearest Neighbors (KNN) dalam Prediksi Properti

K-Nearest Neighbors (KNN) dipilih dalam penelitian ini karena memiliki karakteristik yang sesuai untuk menganalisis data properti yang umumnya bersifat *non-linear*, berbasis fitur numerik, serta dipengaruhi oleh kedekatan spasial maupun karakteristik fisik rumah. Berbeda dengan model parametrik seperti regresi linear, KNN tidak membuat asumsi bentuk hubungan antara fitur dan harga, sehingga lebih fleksibel dalam menangkap pola kompleks yang sering muncul pada data real-estate.

Selain itu, KNN merupakan algoritma *instance-based learning* yang sangat bergantung pada kedekatan antar data. Hal ini relevan dengan konteks harga rumah di Bandung, karena properti yang berlokasi berdekatan atau memiliki karakteristik fisik serupa cenderung memiliki harga yang mirip. Studi oleh Ariyanti dkk. [10] menunjukkan bahwa KNN mampu memberikan kinerja yang kompetitif terhadap model regresi linear, terutama pada dataset dengan pola non-linear, dengan skor R^2 mencapai 49,14%. Temuan ini

mengindikasikan bahwa KNN memiliki kemampuan untuk menangkap variasi harga yang dipengaruhi oleh faktor-faktor lokal.

Di sisi lain, penelitian Putri dkk. [8] melaporkan bahwa performa KNN sangat dipengaruhi oleh parameter jumlah tetangga (k) dan skala fitur. Meskipun demikian, kelemahan ini sekaligus menjadi peluang untuk mengevaluasi dampak proses *hyperparameter tuning* dan *feature scaling* terhadap peningkatan akurasi model. Hal ini menjadi alasan tambahan mengapa KNN layak diteliti lebih mendalam dalam studi komparatif ini, karena proses optimasi dapat memberikan gambaran yang jelas mengenai sejauh mana peningkatan performa KNN dapat dicapai pada konteks pasar properti Bandung.

Dengan demikian, pemilihan KNN bukan hanya sebagai perbandingan sederhana, tetapi juga sebagai perwakilan model non-parametrik yang dapat mengevaluasi bagaimana algoritma berbasis kedekatan bekerja pada dataset real-estate yang kompleks dan berstruktur non-linear.

2.2 Studi Mengenai Model Ansambel (Random Forest & XGBoost)

Model ansambel, yang menggabungkan beberapa model untuk meningkatkan akurasi, telah menunjukkan hasil yang sangat menjanjikan. Sebuah studi [9] secara konsisten membuktikan bahwa Random Forest Regressor menunjukkan kinerja superior dibandingkan *Linear Regression* dan *Decision Tree*. Keunggulan ini terbukti dari metrik evaluasi yang jauh lebih baik, yaitu MAE 18.672,38, MSE 848.269.800, RMSE 29.125,07, dan R² Score mencapai 0,8894.

Keunggulan model ansambel lainnya, XGBoost, juga ditunjukkan dalam penelitian [11]. XGBoost berhasil melampaui model konvensional seperti Regresi Linear (RMSE 0,16530), KNN (RMSE 0,23003), bahkan Random Forest (RMSE 0,14807) dengan mencapai RMSE 0,12745. Studi tersebut juga memanfaatkan SHAP (SHapley Additive exPlanations) untuk menginterpretasikan faktor-faktor yang paling signifikan dalam penentuan harga, memberikan implikasi praktis bagi para pelaku pasar. Studi lain [7] juga mengonfirmasi efektivitas Random Forest dengan tingkat akurasi prediksi mencapai 85,29%, yang diperkuat dengan validasi dari pengujian inferensial dan penerimaan pengguna yang tinggi.

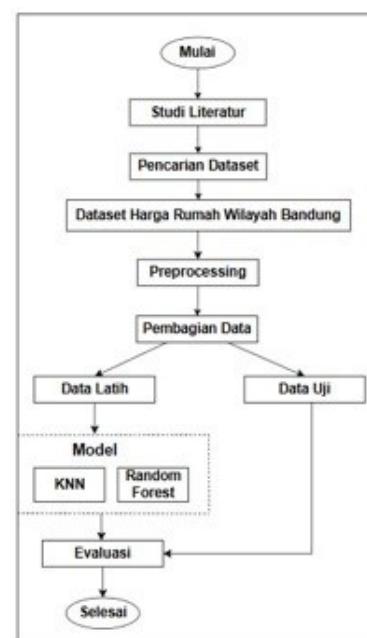
2.3 Sintesis dan Celaht Penelitian

Berdasarkan tinjauan pustaka, model ansambel seperti Random Forest umumnya menunjukkan kinerja prediktif yang lebih unggul dibandingkan model tunggal seperti KNN. Namun, karena performa model sangat bergantung pada konteks geografis dan karakteristik dataset, penelitian yang secara spesifik membandingkan model berbasis instans sederhana (KNN) dengan model ansambel kompleks (Random Forest) untuk pasar properti Bandung yang unik masih terbatas. Oleh karena itu, penelitian ini bertujuan mengisi celah tersebut dengan mengevaluasi dan

mengoptimalkan kedua pendekatan ini untuk menemukan model yang paling andal bagi studi kasus di wilayah Bandung.

III. METODE

Penelitian ini menerapkan metodologi *Cross-Industry Standard Process for Data Mining* (CRISP-DM) untuk membangun model prediksi harga rumah. Metodologi ini menyediakan kerangka kerja terstruktur yang terdiri dari beberapa tahapan utama, mulai dari pemahaman data, persiapan data, pemodelan, hingga evaluasi. Pada tahap pemodelan, penelitian ini secara spesifik membandingkan kinerja dua algoritma machine learning, yaitu K-Nearest Neighbors (KNN) Regressor dan Random Forest (RF) Regressor. Alur penelitian yang diadaptasi dari metodologi CRISP-DM diilustrasikan pada Gambar 1.



Gambar 1. Alur Penelitian Berdasarkan Metodologi CRISP-DM

3.1 Pengumpulan dan Deskripsi Data (Data Collection)

Data yang digunakan dalam penelitian ini merupakan dataset publik berjudul "Daftar Harga Rumah di Kota Bandung" yang diperoleh dari platform Kaggle. Dataset ini diunggah oleh pengguna bernama khaleel347. Dataset dikumpulkan dan diunggah pada Maret 2024, berisi daftar harga rumah beserta atribut-atribut properti dari seluruh kecamatan di Kota Bandung. Secara keseluruhan, dataset mentah terdiri dari 7.609 baris data dan 9 fitur awal. Setelah proses pembersihan dan seleksi awal, ditentukan bahwa terdapat 8 fitur yang relevan untuk pemodelan prediksi harga rumah. Fitur house_name dihapus karena tidak memberikan kontribusi informatif terhadap model dan tidak memiliki nilai prediktif yang signifikan. Deskripsi dataset yang digunakan untuk analisis lebih lanjut dirangkum dalam Tabel 1.

TABEL 1. IDENTIFIKASI DTASET PENELITIAN

| Aspek | Deskripsi |
|--------------------|--|
| Judul Dataset | Daftar harga rumah di bandung |
| Sumber | Kaggle (khaleel347) |
| Target Variabel | Price (Harga Rumah) |
| Jumlah Fitur awal | 9 fitur |
| Jumlah Fitur Akhir | 8 fitur (setelah seleksi) |
| Jumlah Data Awal | 7.609 properti |
| Jumlah Data Akhir | 5.206 properti (setelah preprocessing) |

Dataset ini menyediakan gambaran komprehensif mengenai karakteristik properti di Kota Bandung, mencakup variabel seperti harga rumah (price), jumlah kamar tidur, jumlah kamar mandi, luas tanah, luas bangunan, ketersediaan carport, serta lokasi kecamatan—yang menjadi fitur penting dalam analisis prediksi harga rumah.

3.2 Analisa Data Eksploratif (Exploratory Data Analysis – EDA)

Tahap EDA dilakukan untuk memahami karakteristik fundamental dari dataset melalui analisis statistik dan visualisasi. Proses ini bertujuan untuk mengidentifikasi pola, menemukan anomali, dan memeriksa asumsi sebelum tahap pemodelan. Langkah pertama adalah mengidentifikasi tipe data dari setiap fitur untuk memisahkan antara data kategorikal dan numerik. Rangkuman tipe data disajikan pada Tabel 2.

TABEL 2. DESKRIPSI FITUR DATSEY SEBELUM PREPROCESSING

| No | Fitur | Tipe Data | Keterangan |
|----|----------------|-----------|-------------|
| 1 | Location | object | Kategorikal |
| 2 | Bedroom count | Int64 | Numerik |
| 3 | Bathroom count | Int64 | Numerik |
| 4 | Carport count | Int64 | Numerik |
| 5 | Price | Int64 | Numerik |
| 6 | Land area | Int64 | Numerik |
| 7 | Building area | Int64 | Numerik |

Catatan: Fitur ‘Location’ akan di-encode menjadi 27 fitur binary menggunakan One-Hot Encoding pada tahap preprocessing . Fitur ‘house_name’ tidak ditampilkan karena telah dihapus pada tahap pembersihan data awal.

3.2.1 Analisis Distribusi Fitur

Analisis ini bertujuan untuk memahami sebaran nilai dari setiap fitur:

- Fitur Kategorikal (location): Untuk fitur ini, dilakukan perhitungan frekuensi untuk setiap kategori (kecamatan). Hasilnya kemudian divisualisasikan menggunakan diagram batang (bar chart) untuk mengidentifikasi ada atau tidaknya konsentrasi data pada lokasi tertentu.

- Fitur Numerik: Untuk semua fitur numerik (seperti price, land_area, dll.), dibuat visualisasi histogram dan box plot. Tujuan dari metode ini adalah untuk memeriksa secara visual bentuk distribusi data (apakah simetris atau miring/skewed) dan mengidentifikasi keberadaan pencilan (*outliers*), yaitu nilai-nilai yang berada jauh di luar jangkauan umum data.

3.2.2 Analisis Hubungan Antar Fitur

Langkah ini bertujuan untuk memahami bagaimana fitur-fitur saling berinteraksi, terutama dengan variabel target (*price*).

- Perhitungan Korelasi: Dilakukan perhitungan koefisien korelasi Pearson untuk mengukur kekuatan dan arah hubungan linear antar semua pasangan fitur numerik.
- Visualisasi Korelasi: Hasil dari perhitungan korelasi tersebut kemudian divisualisasikan dalam bentuk matriks korelasi (correlation heatmap). Analisis ini bertujuan untuk mendeteksi potensi multikolinearitas (korelasi tinggi antar fitur prediktor) dan untuk mendapatkan gambaran awal mengenai fitur mana yang memiliki hubungan linear terkuat dengan variabel target (*price*).

Informasi yang diperoleh dari seluruh tahapan EDA ini akan digunakan sebagai dasar untuk merancang strategi pada tahap 3.3 *Data Preprocessing*.

3.3 Pra-pemrosesan Data (Data Preprocessing)

Tahap pra-pemrosesan data bertujuan untuk membersihkan, mentransformasi, dan menyiapkan data mentah agar sesuai dan optimal untuk pemodelan *machine learning*. Berdasarkan temuan dari tahap EDA, beberapa langkah pra-pemrosesan berikut dilakukan secara berurutan:

3.3.1 Seleksi dan Pembersihan Fitur

Pada tahap ini pertama dilakukan yaitu menghapus fitur atau kolom *house_name*, kolom tersebut dihapus karena tidak akan dipakai.

3.3.2 Encoding Data Kategorikal

Model *machine learning* memerlukan input dalam format numerik. Oleh karena itu, fitur kategorikal location perlu diubah menjadi representasi numerik. Dalam penelitian ini, metode *One-Hot Encoding* digunakan. Metode ini mengubah setiap kategori lokasi menjadi fitur biner (0 atau 1) yang terpisah, sehingga model dapat memproses informasi lokasi tanpa membuat asumsi urutan atau peringkat antar lokasi.

3.3.3 Penanganan Pencilan (Outliers)

Outlier adalah data yang berada jauh dari nilai-nilai lama suatu dataset [12]. Apabila terdapat outlier salah satu penanganannya dapat menggunakan *IQR*. *IQR* adalah ukuran

statistic yang digunakan untuk menggambarkan sebaran data dalam dataset [13].

Berdasarkan temuan EDA, beberapa fitur numerik seperti *price*, *land_area*, dan *building_area* menunjukkan distribusi miring (skewed) dan mengandung pencilan ekstrem. Untuk mengurangi pengaruh ekstrem dari pencilan ini, diterapkan metode Interquartile Range (IQR). Data yang berada di luar batas $1.5 * \text{IQR}$ di atas kuartil ketiga (Q3) atau di bawah kuartil pertama (Q1) diidentifikasi dan ditangani dengan teknik capping (menggantinya dengan nilai batas IQR) untuk menjaga distribusi data tetap representatif. Teknik ini dipilih untuk mempertahankan jumlah sampel agar model tidak bias terhadap pengurangan data.

3.3.4 Standarisasi Fitur (*Feature Scaling*)

Langkah terakhir adalah standardisasi fitur, yang bertujuan untuk menyamakan skala semua fitur numerik. Hal ini sangat penting karena KNN sensitif terhadap skala fitur, seluruh fitur numerik dinormalisasi menggunakan StandardScaler, yang mengubah setiap fitur menjadi distribusi dengan mean = 0 dan standar deviasi = 1. StandardScaler diterapkan dalam Pipeline agar proses scaling tetap konsisten selama pelatihan dan validasi.

3.4 Pembagian Data (*Data Splitting*)

Pembagian data merupakan tahap penting dalam proses pembelajaran mesin (*machine learning*), yang bertujuan untuk memisahkan data menjadi dua *subset* utama, yaitu data latih (*training set*) dan data uji (*testing set*).

Setelah data bersih dan siap, dataset dibagi menjadi dua bagian utama:

- Data Latih digunakan oleh algoritma untuk "belajar" menemukan pola yang menghubungkan fitur-fitur dengan harga rumah.
- Data Uji digunakan untuk mengevaluasi seberapa baik model yang telah dilatih dapat melakukan prediksi pada data baru yang belum pernah dilihat sebelumnya.

Dalam penelitian ini, data dibagi dengan proporsi 80% untuk data latih dan 20% untuk data uji. Pembagian ini dilakukan secara acak namun terkontrol dengan menggunakan parameter *random_state* untuk memastikan bahwa hasil pembagian data dapat direproduksi kembali pada eksekusi yang berbeda [14].

3.5 Pemodelan dan Optimalisasi *K-Nearest Neighbour (KNN) Regressor*

3.5.1 Konsep Dasar KNN Regressor

K-Nearest Neighbour (KNN) merupakan salah satu metode *non-parametrik* yang digunakan untuk mengklasifikasikan objek berdasarkan kedekatannya dengan data dalam himpunan pelatihan. Meskipun secara umum KNN lebih dikenal sebagai algoritma klasifikasi, algoritma ini juga dapat

diterapkan untuk tugas *regresi*, yang dikenal sebagai *KNN Regressor*. Pada *regresi* KNN, nilai prediksi ditentukan berdasarkan rata-rata dari k tetangga terdekat [15].

$$y_{t+1} = \frac{1}{k} \sum_{i=1}^k y_i$$

Keterangan :

- $y(t+1)$ = data yang diprediksi
- k = jumlah tetangga terdekat
- y_i = nilai target dari tetangga ke-i.

Algoritma *KNN Regressor* diterapkan dengan melatih model menggunakan data latih yang telah melalui proses pra-proses. Model ini bekerja dengan mencari satu data tetangga terdekat dari data baru (karena nilai k ditetapkan sebesar 1), kemudian melakukan prediksi berdasarkan nilai dari tetangga terdekat tersebut.

3.5.2 Tahapan Pemodelan dan Optimalisasi KNN

Proses pemodelan KNN dalam penelitian ini terdiri dari beberapa langkah sistematis:

- Pemodelan Dasar (Baseline): Pertama, model KNN dasar dilatih menggunakan parameter default dari pustaka Scikit-learn (misalnya, $k=5$). Model ini berfungsi sebagai tolok ukur kinerja awal sebelum dilakukan optimalisasi.
- Konstruksi Pipeline: Mengingat sensitivitas KNN terhadap skala data, proses pemodelan dan optimalisasi digabungkan dalam sebuah Pipeline. Pipeline ini secara berurutan menerapkan langkah StandardScaler (dari tahap pra-pemrosesan) dan pemodelan KNN. Pendekatan ini memastikan bahwa proses standardisasi data selalu diterapkan secara konsisten, terutama selama validasi silang pada proses tuning.
- Optimalisasi dengan Grid Search: Untuk menemukan kombinasi hyperparameter terbaik, digunakan metode GridSearchCV. Metode ini secara sistematis menguji semua kemungkinan kombinasi dari daftar parameter yang telah ditentukan (Tabel 3) untuk menemukan kombinasi yang menghasilkan kinerja terbaik berdasarkan validasi silang.

TABEL 3. RUANG PENCARIAN HYPERPARAMETER UNTUK KNN REGRESSOR

| Hyperparameter | Deskripsi | Rentang Nilai yang Diujii |
|--------------------|-------------------------|----------------------------|
| <i>n_neighbors</i> | Jumlah tetangga (k) | [3, 5, 7, 9, 11, 15, 21] |
| <i>weights</i> | Fungsi bobot | ['uniform', 'distance'] |
| <i>metric</i> | Metrik jarak | ['euclidean', 'manhattan'] |

3.6 Pemodelan dan Optimalisasi Random Forest (RF) Regressor

3.6.1 Konsep Dasar Random Forest Regressor

Random Forest (RF) Regressor adalah metode *ensemble learning* yang membangun sejumlah besar *decision trees* (pohon keputusan) pada berbagai subset data, kemudian menggabungkan hasil prediksi dari semua pohon (biasanya dengan merata-ratakan) untuk menghasilkan prediksi akhir yang lebih akurat dan stabil [16]. Prediksi (\hat{y}) dihitung dengan rumus:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T h_t(x)$$

Keterangan:

- \hat{y} = Nilai prediksi akhir
- T = jumlah pohon
- $h_t(x)$ = hasil prediksi dari pohon ke -i

3.6.2 Tahapan Pemodelan dan Optimalisasi RF

Proses pemodelan untuk Random Forest mengikuti pendekatan dua tahap untuk efisiensi dan ketelitian:

- *Pemodelan Dasar (Baseline)*: Sama seperti KNN, model RF dasar dilatih terlebih dahulu dengan parameter default untuk menetapkan tolok ukur kinerja.
- *Random Search*: Sebagai tahap awal optimalisasi, digunakan RandomizedSearchCV. Metode ini secara acak mencoba sejumlah kombinasi parameter dari rentang pencarian yang luas. Tujuannya adalah untuk secara efisien mempersempit area pencarian ke rentang parameter yang paling menjanjikan tanpa harus menguji setiap kombinasi.
- *Grid Search*: Setelah rentang parameter yang potensial diidentifikasi melalui Random Search, pencarian yang lebih mendalam dan sistematis dilakukan menggunakan GridSearchCV. Metode ini menguji semua kombinasi secara menyeluruh pada rentang parameter yang lebih sempit untuk menemukan kombinasi optimal yang presisi. Ruang pencarian parameter untuk kedua tahap ini dirangkum dalam Tabel 4.

TABEL 4. RUANG PENCARIAN HYPERPARAMETER UNTUK RANDOM FOREST REGRESSOR

| Hyperparameter | Deskripsi | Rentang Nilai yang Diuji |
|--------------------------|--------------------------------------|--------------------------|
| <i>n_estimators</i> | Jumlah pohon dalam forest | [100, 200, 500, 1000] |
| <i>Max_depth</i> | Kedalaman maksimum pohon | [10, 20, 30, None] |
| <i>min_samples_split</i> | Metrik jarak | [2, 5, 10] |
| <i>Min_samples_leaf</i> | Jumlah minimum sampel pada leaf node | [1, 2, 4] |
| <i>max_features</i> | Jumlah fitur untuk dipertimbangkan | ['auto', 'sqrt'] |

3.7 Evaluasi Model

Setelah model dilatih dan dioptimalkan, tahap selanjutnya adalah mengevaluasi kinerja dan keandalannya dalam melakukan prediksi. Evaluasi dilakukan pada data uji (test set), yang merupakan data yang belum pernah dilihat oleh model selama proses pelatihan. Tujuan dari tahap ini adalah untuk mengukur seberapa akurat prediksi model dibandingkan dengan nilai aktual. Dalam penelitian ini, kinerja model regresi diukur menggunakan tiga metrik standar, yaitu *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan *R-Squared* (R^2).

3.7.1 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) adalah metrik yang mengukur rata-rata dari selisih absolut (nilai mutlak) antara nilai prediksi dan nilai aktual. MAE memberikan gambaran tentang besaran rata-rata kesalahan prediksi dalam satuan yang sama dengan variabel target (Rupiah). Semakin kecil nilai MAE, semakin baik kinerja model karena menunjukkan bahwa rata-rata kesalahan rendah [17].

Formula MAE dihitung sebagai berikut:

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t|$$

Keterangan :

- n = jumlah total data
- y_t = nilai aktual pada observasi ke-t
- \hat{y}_t = nilai prediksi pada observasi ke-t

3.7.2 Mean Squared Error (MSE)

Mean Squared Error (MSE) menghitung rata-rata dari kuadrat selisih antara nilai prediksi dan nilai aktual. Dengan mengkuadratkan selisihnya, MSE memberikan "hukuman" yang lebih besar terhadap kesalahan prediksi yang besar. Metrik ini sangat berguna untuk melihat apakah model memiliki penciran prediksi yang sangat buruk. Sama seperti MAE, nilai MSE yang lebih rendah menunjukkan kinerja model yang lebih baik [17].

Formula MSE dihitung sebagai berikut:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Keterangan :

- n = Jumlah total data pada data uji
- y_i = Nilai harga rumah aktual pada data ke-i
- \hat{y}_i = Nilai harga rumah prediksi pada data ke-i

3.7.3 R-Squared (R^2 atau Koefisien Determinasi)

R^2 adalah metrik statistik yang mengukur seberapa besar persentase variasi dari variabel target (harga rumah) yang dapat dijelaskan oleh fitur-fitur (prediktor) dalam model. Nilai R^2 berkisar antara 0 hingga 1. Nilai yang semakin mendekati 1 menunjukkan bahwa model mampu menjelaskan sebagian besar variabilitas data, yang mengindikasikan kinerja model yang lebih baik. Sebaliknya, nilai yang mendekati 0 menunjukkan bahwa model tidak mampu menjelaskan variasi data dengan baik.

Formula R^2 dihitung sebagai berikut:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

Keterangan :

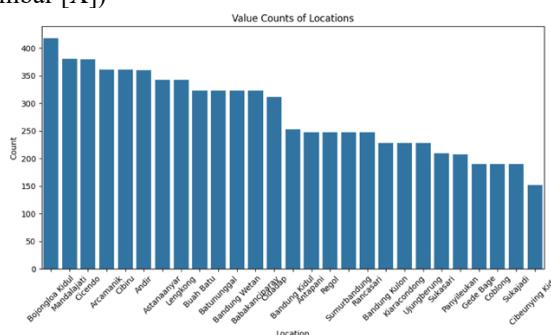
- y_i = Nilai harga rumah aktual pada data ke- i
- \hat{y}_i = Nilai harga rumah prediksi pada data ke- i
- \bar{y} = Nilai rata-rata dari harga rumah aktual

IV. HASIL DAN PEMBAHASAN

Bab ini secara komprehensif menguraikan hasil dari seluruh tahapan penelitian serta pembahasannya. Analisis dimulai dengan menyajikan wawasan kunci yang diperoleh dari tahap analisis data eksploratif (EDA) dan pra-pemrosesan. Fokus utama kemudian beralih pada penyajian hasil kinerja kuantitatif dari model K-Nearest Neighbors (KNN) dan Random Forest (RF) yang telah dioptimalkan. Bab ini ditutup dengan diskusi mendalam yang membandingkan kedua model, menganalisis faktor-faktor yang memengaruhi performa, serta membahas implikasi praktis dan keterbatasan dari penelitian ini.

4.1 Hasil Analisis Data Eksploratif (EDA) dan Data Preprocessing

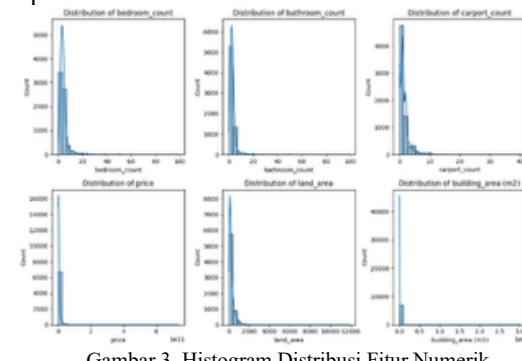
Analisis data eksploratif (EDA) terhadap dataset awal mengungkap beberapa karakteristik kunci yang menjadi landasan untuk tahap pra-pemrosesan. Ditemukan bahwa distribusi properti tidak merata, dengan konsentrasi di beberapa kecamatan tertentu seperti Bojongloa Kidul (Gambar [X])



Gambar 2. Distribusi Frekuensi Properti Berdasarkan Kecamatan di Kota Bandung

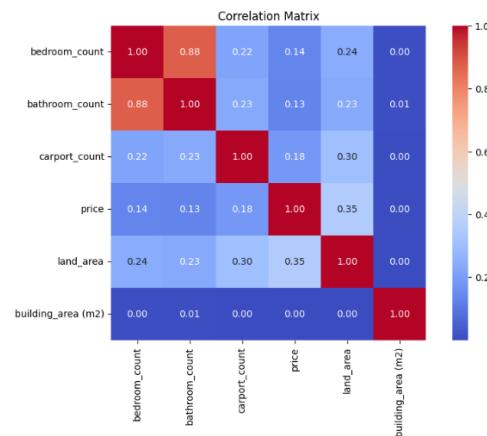
Fitur kategorikal *location* dianalisis untuk memahami sebaran geografis data properti. Dari total 7.609 data, teridentifikasi 27 lokasi kecamatan yang unik. Analisis frekuensi pada Gambar 2 menunjukkan bahwa data tidak terdistribusi merata, melainkan terkonsentrasi pada beberapa lokasi tertentu.

Selain itu, fitur numerik utama seperti *price*, *land_area*, *building_area* pada gambar 3 menunjukkan distribusi yang sangat miring ke kanan (*right-skewed*) dan mengandung banyak pencilan



Gambar 3. Histogram Distribusi Fitur Numerik

Untuk mengukur kekuatan dan arah hubungan linear antara fitur-fitur numerik, dibuatlah sebuah matriks korelasi yang divisualisasikan sebagai *heatmap* pada Gambar 4.



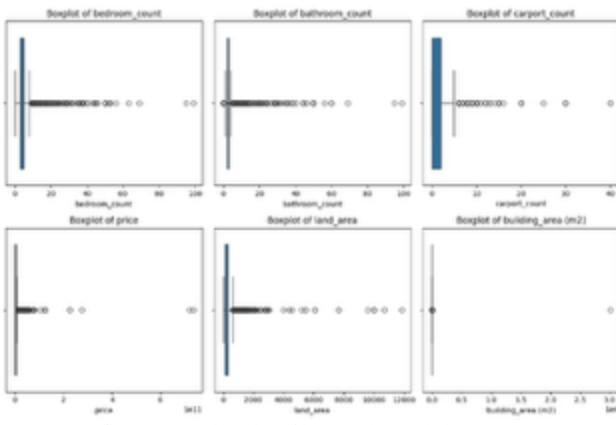
Gambar 4. Matriks Korelasi Antar Fitur Numerik

Analisis heatmap korelasi ini memberikan beberapa temuan krusial:

- Multikolinearitas: Terdapat korelasi positif yang sangat kuat (0.88) antara *bedroom_count* dan *bathroom_count*. Nilai korelasi yang tinggi ini menandakan adanya multikolinearitas, yang perlu diperhatikan dalam pemodelan.
- Korelasi dengan Target Harga: Fitur *land_area* menunjukkan korelasi positif tertinggi dengan *price* (0.35), yang menegaskan bahwa luas tanah adalah faktor penting dalam penentuan harga.

Temuan-temuan dari keseluruhan proses EDA ini akan menjadi landasan utama untuk menentukan langkah-langkah yang akan diambil pada tahap Data Preprocessing

Tahap pertama yang dilakukan dalam preprosesing data yaitu menghapus salah satu kolom yang tidak terpakai, yaitu kolom *house name*. Lalu mengubah tipe data pada kolom *location* menjadi numerik. Pengecekan *outlier* juga perlu dilakukan pada tahap *pre-processing* ini.

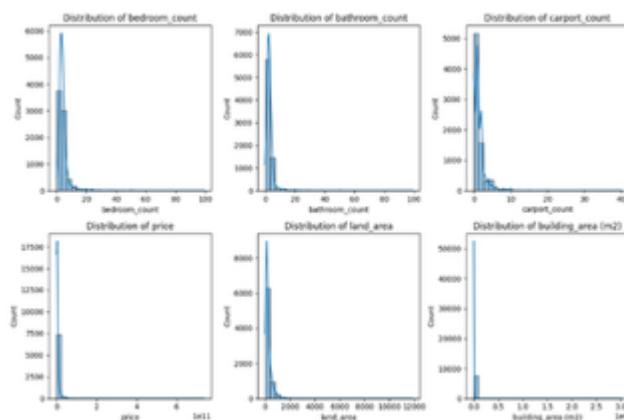


Gambar 5. Box Plot Identifikasi outlier Sebelum Penanganan

Pada *box plot* yang ditampilkan, semua kolom mempunyai *outlier*, diantaranya *bedroom count*, *bathroom count*, *carpot count*, *price*, *land area*, *building area(m2)*. Ke-enam kolom tersebut selanjutnya dilakukan penanganan dan penghapusan *outlier* dengan menggunakan *Inter-Quartile Range* (IQR). Setelah penaganan *outlier* dilakukan ukuran dataset berubah menjadi 7 kolom 5206 baris.

TABEL 5. HASIL PENANGANAN OUTLIER

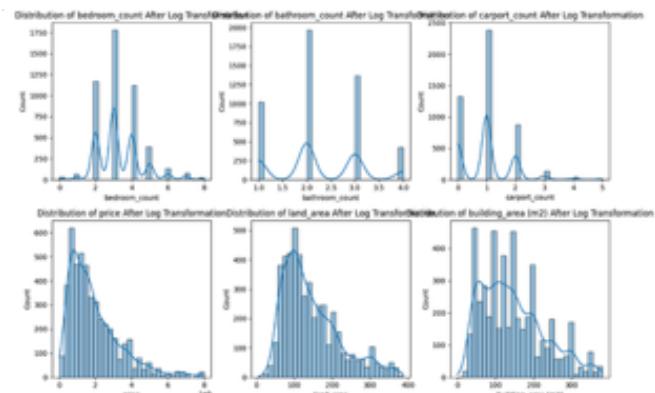
| SEBELUM | SESUDAH |
|---------|---------|
| 7 | 7 |
| 7610 | 5206 |



Gambar 6. Distribusi Fitur Numerik Setelah Penanganan Outlier

Distribusi data setelah dilakukan log transformasi menunjukkan peningkatan dalam persebaran dan normalitas data pada seluruh atribut. Atribut *bedroom_count*,

bathroom_count, dan *carport_count* yang sebelumnya memiliki penciran ekstrem kini tampak lebih seimbang, meskipun masih terlihat beberapa puncak distribusi akibat dominasi nilai-nilai tertentu. Distribusi *price* yang awalnya sangat miring ke kanan kini menjadi lebih simetris, menandakan keberhasilan log transformasi dalam mengurangi *skewness*. Atribut *land_area* dan *building_area(m2)* juga mengalami penyebaran yang lebih baik dan mengarah ke distribusi yang mendekati normal. Transformasi ini penting untuk meningkatkan performa algoritma prediktif, terutama yang sensitif terhadap skala dan distribusi data seperti *K-Nearest Neighbors* dan regresi berbasis pohon.



Gambar 7. Distribusi Fitur Numerik Setelah Standardisasi menggunakan StandardScaler

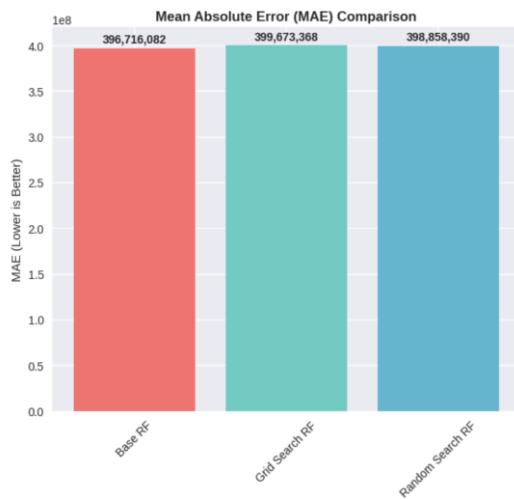
Setelah data bersih dan ditransformasi, langkah selanjutnya adalah membaginya menjadi data latih dan data uji. Sesuai dengan proporsi 80:20 yang ditetapkan dalam metodologi, dataset final dibagi menjadi 4.164 sampel untuk data latih dan 1.042 sampel untuk data uji. Terakhir, untuk memastikan semua fitur memiliki skala yang seragam sebuah syarat penting untuk algoritma KNN—dilakukan proses standardisasi menggunakan StandardScaler. Proses ini diterapkan dengan mempelajari parameter (rata-rata dan standar deviasi) dari data latih, yang kemudian digunakan untuk mentransformasi baik data latih maupun data uji secara konsisten. Dengan demikian, dataset kini telah sepenuhnya siap untuk tahap pemodelan.

4.2 Hasil Kinerja Model

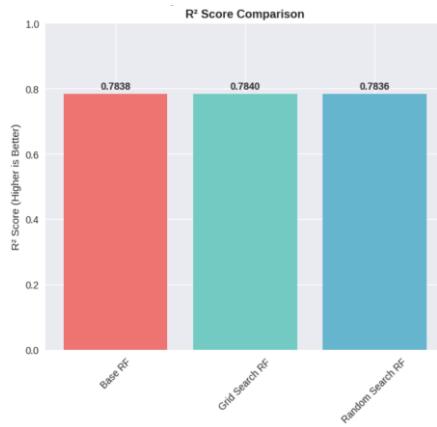
Setelah data disiapkan melalui serangkaian tahap prapemrosesan, langkah selanjutnya adalah melatih dan mengevaluasi model K-Nearest Neighbors (KNN) dan Random Forest (RF) sesuai dengan metodologi yang diuraikan pada Bab 3. Bagian ini menyajikan hasil kinerja kuantitatif dari model dasar (*baseline*) sebelum optimalisasi, hasil dari proses optimalisasi *hyperparameter*, serta perbandingan kinerja akhir antara kedua model yang telah dioptimalkan. Evaluasi dilakukan pada data uji menggunakan metrik *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan *R-Squared* (R^2).

4.2.1 Analisis Kinerja Model Random Forest Regressor

Analisis kinerja pertama difokuskan pada model Random Forest (RF) Regressor, di mana model dasar (baseline) dengan parameter default ditetapkan sebagai tolok ukur awal. Setelah itu, dilakukan proses optimalisasi hyperparameter secara intensif menggunakan *RandomizedSearchCV* dan *GridSearchCV*. Namun, hasil evaluasi menunjukkan bahwa model dasar RF sudah memiliki kinerja yang sangat kuat. Gambar 8 dan Gambar 9 di bawah ini memvisualisasikan perbandingan metrik Mean Absolute Error (MAE) dan R² Score pada ketiga tahap pemodelan.



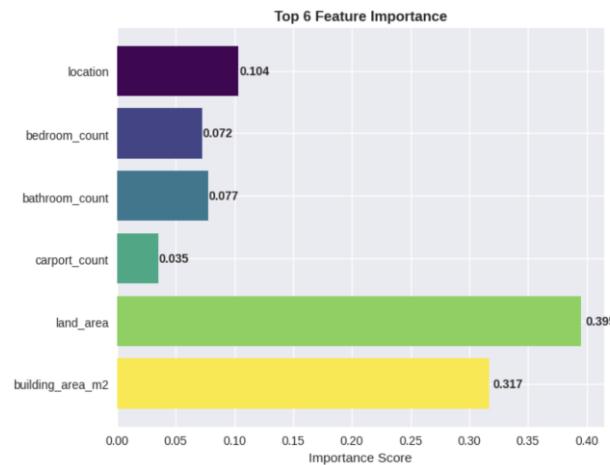
Gambar 8. Perbandingan MAE Model Random Forest pada Tiga Tahap Pemodelan



Gambar 9. Perbandingan R² Score Model Random Forest pada Tiga Tahap Pemodelan

Terlihat bahwa upaya optimalisasi tidak memberikan peningkatan kinerja yang signifikan; model hasil *Grid Search* hanya sedikit meningkatkan R² Score menjadi 0.7840 dari 0.7838, sementara nilai MAE-nya justru sedikit lebih tinggi. Temuan ini, mengindikasikan bahwa parameter default dari algoritma Random Forest sudah sangat efektif untuk dataset ini. Untuk memahami faktor yang paling dominan dalam prediksi model, dilakukan pula analisis tingkat kepentingan fitur (*feature importance*). Seperti yang divisualisasikan pada Gambar 10, hasilnya secara jelas menunjukkan bahwa luas

tanah (land_area) dengan bobot 39.5% dan luas bangunan (building_area_m2) dengan bobot 31.7% adalah dua prediktor paling berpengaruh. Kedua fitur ini secara gabungan menyumbang lebih dari 71% dari total kekuatan prediksi model. Faktor penting berikutnya adalah lokasi (location) sebesar 10.4%, yang menegaskan bahwa nilai properti di Bandung sangat ditentukan oleh ukuran fisik properti dan lokasinya.

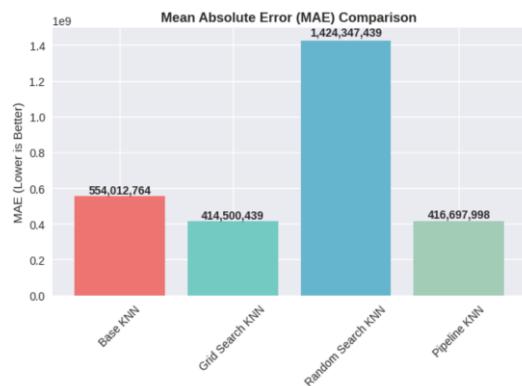


Gambar 10. Tingkat Kepentingan Fitur pada Model Random Forest

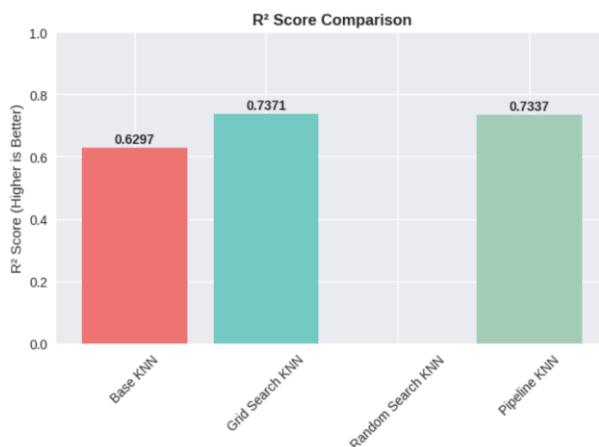
4.2.2 Analisis Kinerja Model K-Nearest Neighbors (KNN) Regressor

Analisis selanjutnya difokuskan pada model K-Nearest Neighbors (KNN) Regressor. Model dasar (baseline) KNN, yang dilatih tanpa optimalisasi, menunjukkan kinerja awal yang moderat dengan R² Score sebesar 0.6297 dan MAE sekitar Rp 554 juta. Kinerja model ini sangat bergantung pada hyperparameter dan skala fitur, sehingga proses optimalisasi menjadi langkah yang krusial. Melalui *GridSearchCV* yang diterapkan dalam sebuah Pipeline dengan *StandardScaler*, berhasil ditemukan konfigurasi optimal yaitu dengan menggunakan 17 tetangga (*n_neighbors*), pembobotan berbasis jarak (*weights='distance'*), dan metrik Manhattan distance (*p=1*).

Penerapan parameter optimal ini menghasilkan peningkatan kinerja yang sangat signifikan, seperti yang divisualisasikan pada Gambar 11 dan Gambar 12. Model KNN yang telah dioptimalkan berhasil mencapai R² Score sebesar 0.7371 dan menekan MAE hingga Rp 414,5 juta, atau turun lebih dari 25% dibandingkan model dasarnya. Rangkuman perbandingan kuantitatif yang drastis ini disajikan pada Tabel [Y]. Temuan ini menegaskan bahwa untuk algoritma seperti KNN, proses hyperparameter tuning dan standardisasi fitur adalah langkah yang mutlak diperlukan untuk mencapai performa yang kompetitif.



Gambar 11. Perbandingan MAE Model KNN Dua Tahap Pemodelan



Gambar 12. Perbandingan R² Score Model KNN pada Dua Tahap Pemodelan

4.3 Pembahasan dan Perbandingan Model

Setelah menganalisis kinerja masing-masing model secara terpisah, tahap ini bertujuan untuk membandingkan secara langsung performa akhir dari model K-Nearest Neighbors (KNN) dan Random Forest (RF) yang telah dioptimalkan. Perbandingan ini dilakukan untuk menentukan model mana yang paling unggul dan andal untuk tugas prediksi harga rumah di Bandung.

4.3.1 Perbandingan Kuantitatif Kinerja Model

Evaluasi pada data uji secara jelas menunjukkan bahwa model Random Forest Regressor secara konsisten lebih unggul dibandingkan K-Nearest Neighbors Regressor pada semua metrik yang diukur. Tabel 6 di bawah ini menyajikan perbandingan langsung kinerja kedua model.

TABEL 6. PERBANDINGAN KINERJA MODEL KNN DAN RANDOM FOREST

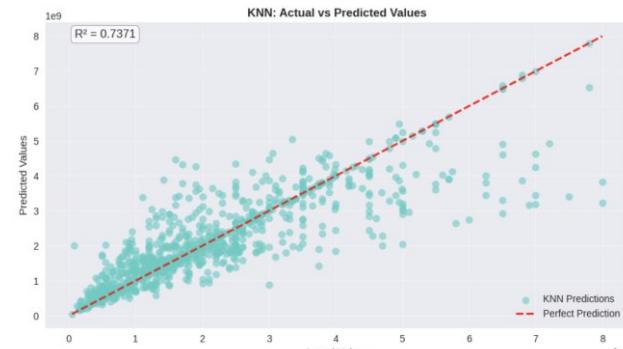
| Metrik | Model KNN (Tuned) | Model RF Base | Pemenang |
|----------|-----------------------|-----------------------|----------|
| MAE | Rp 414,5 Juta | Rp 399,7 Juta | RF Base |
| MSE | 5.58×10^{17} | 4.59×10^{17} | RF Base |
| R² Score | 0.7371 | 0.7838 | RF Base |

Dari tabel tersebut, terlihat bahwa Random Forest berhasil memenangkan ketiga metrik utama. Secara spesifik, model Random Forest menghasilkan Mean Absolute Error (MAE) sekitar Rp 14,8 juta (3.58%) lebih rendah dibandingkan KNN. Perbedaan paling signifikan terlihat pada metrik MSE, di mana RF 17.85% lebih rendah, yang mengindikasikan bahwa RF jauh lebih baik dalam menghindari kesalahan prediksi yang sangat besar. Selain itu, R² Score RF yang mencapai 0.7838 menunjukkan kemampuannya untuk menjelaskan 78.4% variasi harga rumah, hampir 6% lebih tinggi dibandingkan KNN.

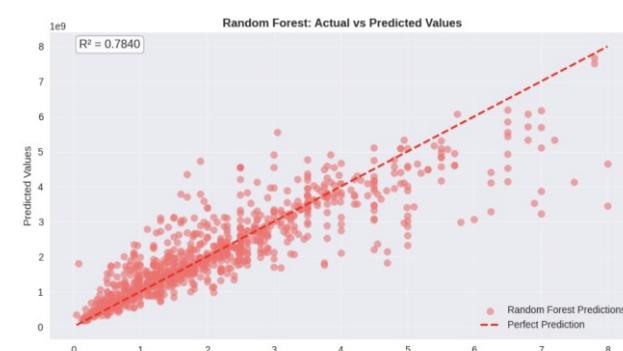
4.3.2 Analisis Kualitatif Pola Prediksi

Analisis kualitatif pada tingkat prediksi individual memberikan wawasan yang lebih mendalam di luar metrik agregat. Meskipun Random Forest unggul secara keseluruhan dengan memberikan prediksi yang lebih akurat pada 560 dari 1042 sampel uji (dibandingkan 471 untuk KNN), pola keunggulannya menjadi lebih jelas saat dianalisis secara visual.

Pola prediksi dari kedua model dapat diamati melalui scatter plot yang membandingkan nilai aktual dengan nilai prediksi, seperti yang ditunjukkan pada Gambar 13 dan Gambar 14.



Gambar 13. Perbandingan Nilai Aktual vs. Prediksi (Model KNN)



Gambar 14. Perbandingan Nilai Aktual vs. Prediksi (Model Random Forest)

Pada Gambar 13, terlihat bahwa Sebaran titik yang dispersed menunjukkan variance prediksi tinggi. Model KNN mengalami deviasi signifikan pada properti berharga tinggi (>Rp 3 miliar), dengan kesalahan prediksi mencapai Rp 1,5-

2 miliar, mengindikasikan sensitivitas model terhadap outlier dan keterbatasan generalisasi pada segmen premium. Hal ini mengindikasikan bahwa prediksinya lebih konsisten dan memiliki tingkat kesalahan yang relatif stabil di berbagai rentang harga. Sebaran titik yang lebih rapat di sekitar garis diagonal mengindikasikan konsistensi prediksi yang superior. Random Forest menunjukkan stabilitas prediksi yang lebih baik di semua rentang harga.

Sebaliknya, pada Gambar 14, Sebaran titik yang lebih rapat dan linear di sekitar garis diagonal menunjukkan konsistensi prediksi superior. Random Forest menunjukkan stabilitas tinggi di semua rentang harga, dengan deviasi yang relatif merata dari properti murah hingga premium, membuktikan kemampuan generalisasi yang lebih baik dibandingkan KNN.

Pola ini terkonfirmasi pada salah satu sampel uji dengan harga aktual Rp 2,5 Miliar. Model Random Forest memberikan prediksi sebesar Rp 3,04 Miliar, sementara prediksi KNN jauh meleset di angka Rp 3,7 Miliar. Kasus ini secara nyata mengilustrasikan keunggulan RF dalam melakukan generalisasi pada properti bernilai tinggi. Secara visual, bukti ini memperkuat temuan kuantitatif bahwa Random Forest tidak hanya lebih akurat, tetapi juga lebih stabil dan andal di seluruh spektrum harga properti.

V. KESIMPULAN

Berdasarkan analisis dan evaluasi yang telah dilakukan, penelitian ini menyimpulkan bahwa model Random Forest (RF) Regressor secara signifikan lebih unggul dibandingkan K-Nearest Neighbors (KNN) Regressor untuk memprediksi harga rumah di Bandung. Dengan R^2 Score mencapai 0.7838, Random Forest terbukti lebih akurat dan andal. Hasil ini juga mengungkap bahwa proses hyperparameter tuning merupakan langkah krusial yang mampu meningkatkan kinerja KNN secara drastis, namun hanya memberikan dampak minimal pada model Random Forest yang sudah sangat robust dengan parameter dasarnya. Analisis lebih lanjut menunjukkan bahwa luas tanah, luas bangunan, dan lokasi merupakan tiga faktor paling dominan yang memengaruhi harga properti dalam dataset ini.

Kontribusi utama dari penelitian ini adalah menyediakan model prediksi harga rumah berbasis pembelajaran mesin yang tervalidasi untuk konteks pasar properti Bandung. Penelitian ini memberikan pemahaman empiris mengenai perbandingan dua algoritma populer (KNN dan Random Forest) serta mengidentifikasi fitur-fitur paling berpengaruh terhadap harga rumah. Selain itu, penelitian ini menghasilkan kerangka kerja praproses data yang sistematis—meliputi penanganan outlier dengan IQR, encoding fitur kategorikal menggunakan One-Hot Encoding, dan standardisasi fitur—yang dapat direplikasi untuk kasus serupa. Dari sisi praktis, model Random Forest yang dibangun berpotensi menjadi alat bantu estimasi harga properti yang objektif dan dapat meningkatkan transparansi dalam transaksi real estate di Bandung. Dari sisi teoretis, penelitian ini memperkuat bukti empiris mengenai keunggulan metode ensemble dalam tugas

regresi kompleks dan menegaskan urgensi praproses data yang tepat.

Meskipun demikian, penelitian ini memiliki beberapa keterbatasan yang perlu diakui. Keterbatasan utama terletak pada dataset yang tidak mencakup variabel kualitatif penting seperti kondisi fisik properti atau riwayat renovasi, yang dapat memengaruhi akurasi prediksi. Selain itu, meskipun unggul, model Random Forest masih menunjukkan rata-rata kesalahan (MAE) yang cukup signifikan (sekitar Rp 399,7 Juta) dan performanya cenderung menurun pada segmen properti bernilai sangat tinggi (rumah mewah) yang jumlahnya terbatas dalam data.

Untuk penelitian di masa mendatang, disarankan untuk melakukan pengayaan dataset dengan menyertakan fitur-fitur kualitatif yang disebutkan sebelumnya. Eksplorasi algoritma ensemble yang lebih canggih seperti XGBoost atau LightGBM juga dapat dilakukan untuk melihat potensi peningkatan akurasi.

DAFTAR PUSTAKA

- [1] E. S. Lestari and I. Astuti, "Penerapan Random Forest Regression Untuk Memprediksi Harga Jual Rumah Dan Cosine Similarity Untuk Rekomendasi Rumah Pada Provinsi Jawa Barat," *J. Ilm. FIFO*, vol. 14, no. 2, p. 131, 2022, doi: 10.22441/fifo.2022.v14i2.003.
- [2] Savitri, N. F., and Nasrudin, N. (2023, November). Peramalan indeks harga properti residensial di kota Bandung tahun 2023. *Jurnal Kebijakan Pembangunan Daerah*, 7(2), 140–157.
- [3] L. Somantri, "Land Price Mapping in the Northern Suburbs of Bandung City West Java Province Indonesia," *Forum Geografi*, vol. 34, no. 1, pp. 26–40, Jul. 2020, doi: 10.23917/forgeo.v34i1.10412
- [4] R. Khoirudin and M. L. A. Kurniawan, "A time-varying of property residential price in Indonesia: a VAR approach." *Jurnal Ekonomi & Studi Pembangunan*, vol. 24, no. 1, Apr. 2023
- [5] E. Fitri, "Analisis Perbandingan Metode Regresi Linier, Random Forest Regression dan Gradient Boosted Trees Regression Method untuk Prediksi Harga Rumah," *J. Appl. Comput. Sci. Technol.*, vol. 4, no. 1, pp. 58–64, 2023, doi: 10.52158/jacost.v4i1.491.
- [6] F. Hidayah, S. J. Angesti, and Y. P. Widayastuti, "Prediksi Harga Rumah di Boston Menggunakan Metode Linear Regression, SVR, Decision Tree dan Random Forest Regression," pp. 1–9, 2024.
- [7] N. Nuris, "Analisis Prediksi Harga Rumah Pada Machine Learning Metode Regresi Linear," *Explore*, vol. 14, no. 2, pp. 108–112, 2024, doi: 10.35200/ex.v14i2.123.
- [8] V. Ariyani, P. Putri, A. B. Prasetijo, and D. Eridani, "Perbandingan Kinerja Algoritme Naïve Bayes Dan K-Nearest Neighbor (Knn) Untuk Prediksi Harga Rumah," *J. Ilm. Tek. Elektro*, vol. 24, no. 2, pp. 162–171, 2022, [Online]. Available: <https://ejournal.undip.ac.id/index.php/transmisi>
- [9] D. Nyoman, M. Cahyani, N. Putu, and K. Indah, "Comparison Of Decision Tree, Linear Regression, and Random Forest Regressor Models for Predicting House Prices Desak Nyoman Mulya Cahyani et al," *J. Ilm. Merpati*, vol. 12, no. 1, pp. 62–71, 2024.
- [10] Nadia Putri Ariyanti, Agung Triayudi, and Ratih Titi Komala Sari, "Analysis of K-NN Algorithm and Linear Regression to Predict House Prices in Jabodetabek," *SaNa J. Blockchain, NFTs Metaverse Technol.*, vol. 2, no. 1, pp. 65–71, 2024, doi: 10.58905/sana.v2i1.265.
- [11] D. E. Aniobi, C. O. Ochuba, and S. B. Nguideen, "House Price Prediction: Comparative Analysis of Regression-Based Machine Learning Algorithms," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 11, no. 10, pp. 1550–1557, 2023, doi: 10.22214/ijraset.2023.56232.

- [12] P. R. Sihombing, D. A. Sunarjo, Y. Paulus, and A. C. Yuda, "Jurnal Ekonomi dan Statistik Indonesia," vol. 2, no. 3, pp. 307–315, 2022, doi: 10.11594/jesi.02.03.07.
- [13] S. M. Faradisa, T. D. Nugrahadi, Muliadi, I. Budiman, and D. Kartini, "Implementasi IQR-SMOTE Untuk Mengatasi Ketidakseimbangan Kelas Pada Klasifikasi Diabetes menggunakan K-Nearest Neighbors," vol. 15, pp. 48–60, 2021.
- [14] S. T. Rizaldi and M. Mustakim, "Perbandingan Teknik Pembagian Data untuk Klasifikasi Sarana Akses Air pada Algoritma K-Nearest Neighbor dan Naïve Bayes Classifier," *Semin. Nas. Teknol. Informasi, Komun. dan Ind.* 12, pp. 130–137, 2020.
- [15] D. S. Seruni, M. T. Furqon, and R. C. Wihandika, "Sistem Prediksi Pertumbuhan Jumlah Penduduk Kota Malang menggunakan Metode K-Nearest Neighbor Regression," *Sist. Prediksi Pertumbuhan Jumlah Penduduk Kota Malang menggunakan Metod. K-Nearest Neighbor Regres.*, vol. 4, no. 4, pp. 1075–1082, 2020.
- [16] Warjiyono, A. Nur Rais, I. Alfarobi, S. Wira Hadi, and W. Kurniawan, "Analisa Prediksi Harga Jual Rumah Menggunakan Algoritma Random Forest Machine Learning," *JURSISTEKNI (Jurnal Sist. Inf. dan Teknol. Informasi)*, vol. 6, no. 2, pp. 416–423, 2024.
- [17] N. N. Sari, T. T. Anisah, and R. Fitriani, "Implementasi Machine Learning untuk Prediksi Harga Laptop Menggunakan Algoritma Regresi Linear Berganda Machine Learning Implementation for Laptop Price Prediction Using Multiple Linear Regression Algorithm," vol. 14, pp. 162–177, 2024.