FESMARO 2025

DOKUMEN METODOLOGI

Festival Kreativitas Mahasiswa Elektro dan Informatika (FESMARO) 2025

Tim Bubadibako

- Rizky Febrian Dwi Putra (3022210026)
- Fahmi Abdullah Muslimin (3022210009)
- Balya Badar Syah (3022210007

Universitas Internasional Semen Indonesia

METODOLOGI

A. Data Source

Dataset yang digunakan dalam project ini berasal dari diperoleh dari : *https://drive.google.com/drive/folders/1M8Q5SDmWcBcGRIaF1OkLCYivvUI46Mcc?usp=sharing*. Dataset ini berasal dari "DataCo Smart Supply Chain Dataset" milik DataCo Global, yang mencakup informasi rantai pasok meliputi aktivitas provisioning, produksi, penjualan, dan distribusi komersial untuk produk seperti pakaian, perlengkapan olahraga, dan elektronik. Data ini mencakup detail transaksi, pengiriman, pelanggan, serta pesanan, yang memungkinkan analisis prediktif seperti optimasi logistik, prediksi permintaan, dan klasifikasi perilaku pelanggan. Berikut penjelasan dari tiap fitur pada dataset ini:

Nama Kolom	Penjelasan
Type	Type of transaction made
Days for shipping (real)	Actual shipping days of the purchased product
Days for shipment (scheduled)	Days of scheduled delivery of the purchased product
Benefit per order	Earnings per order placed
Sales per customer	Total sales per customer made per customer
Delivery Status	Delivery status of orders: Advance shipping, Late delivery, Shipping canceled, Shipping on time

Late_delivery_risk	Categorical variable that indicates if sending is late (1), it is not late (0)
Category Id	Product category code
Category Name	Description of the product category
Customer City	City where the customer made the purchase
Customer Country	Country where the customer made the purchase
Customer Email	Customer's email
Customer Fname	Customer first name
Customer Id	Customer ID
Customer Lname	Customer last name
Customer Password	Masked customer key
Customer Segment	Types of Customers: Consumer, Corporate, Home Office
Customer State	State to which the store where the purchase is registered belongs

Customer Street	Street to which the store where the purchase is registered belongs
Customer Zipcode	Customer Zipcode
Department Id	Department code of store
Department Name	Department name of store
Latitude	Latitude corresponding to location of store
Longitude	Longitude corresponding to location of store
Market	Market to where the order is delivered: Africa, Europe, LATAM, Pacific Asia, USCA
Order City	Destination city of the order
Order Country	Destination country of the order
Order Customer Id	Customer order code
order date (DateOrders)	Date on which the order is made
Order Id	Order code

Order Item Cardprod Id	Product code generated through the RFID reader
Order Item Discount	Order item discount value
Order Item Discount Rate	Order item discount percentage
Order Item Id	Order item code
Order Item Product Price	Price of products without discount
Order Item Profit Ratio	Order Item Profit Ratio
Order Item Quantity	Number of products per order
Sales	Value in sales
Order Item Total	Total amount per order
Order Profit Per Order	Order Profit Per Order
Order Region	Region of the world where the order is delivered (e.g., Southeast Asia, Western Europe, etc.)
Order State	State of the region where the order is delivered

Order Status	Order Status: COMPLETE, PENDING, CLOSED, PENDING_PAYMENT, CANCELED, etc.
Product Card Id	Product code
Product Category Id	Product category code
Product Description	Product Description
Product Image	Link of visit and purchase of the product
Product Name	Product Name
Product Price	Product Price
Product Status	Status of the product stock: 1 (not available), 0 (available)
Shipping date (DateOrders)	Exact date and time of shipment
Shipping Mode	Shipping modes: Standard Class, First Class, Second Class, Same Day

B. Alat dan Teknologi

Bahasa Pemrograman: Python
Library dan Tools:

- Data Processing: Pandas, NumPy
- Machine Learning: scikit-learn, LightGBM, XGBoost, CatBoost, ElasticNet
- Optimasi Hyperparameter: Hyperopt
- Visualisasi: Matplotlib, Seaborn

Platform Pengembangan: Google Collab

C. Data Preprocessing

a. **Pemeriksaan dan Data Wrangling**

Dataset yang digunakan memiliki 180.519 entri dan 53 kolom dengan berbagai tipe data (object, int64, dan float64) yang mencakup informasi transaksi, data pelanggan, detail produk, serta informasi geografis seperti latitude dan longitude. Data diambil dari file CSV menggunakan library pandas yang memungkinkan integrasi data ke dalam lingkungan analisis dengan mudah. Setelah data dimuat, dilakukan pemeriksaan awal dengan menampilkan beberapa baris data, melihat informasi struktur dan statistik deskriptif, serta mengidentifikasi potensi masalah seperti nilai kosong/blank yang mencapai 336.209 sel. Meski begitu, tidak ditemukan adanya duplikasi entri. Statistik deskriptif ini memberikan gambaran mengenai nilai rata-rata, standar deviasi, dan rentang nilai pada fitur numerik, sehingga memudahkan pemahaman karakteristik data sebelum dilanjutkan ke proses pembersihan dan pengembangan model.

b. **Cleaning Data**

Proses pembersihan dimulai dengan mengatasi masalah format angka, terutama pada angka yang menggunakan tanda koma sebagai pemisah ribuan. Fungsi khusus dibuat untuk mengkonversi nilai string menjadi tipe numerik yang benar. Selain itu, kolom-kolom yang tidak relevan seperti Order Zipcode dan Product Description dihapus guna menyederhanakan dataset. Selanjutnya, data diaggregasi berdasarkan Order Id dengan menerapkan fungsi agregasi seperti penjumlahan, rata-rata, dan modus untuk menyusun ringkasan tiap order, sehingga informasi yang tersebar pada beberapa baris dapat digabungkan secara representatif.

c. Feature Engineering

Dalam tahap feature engineering, dilakukan beberapa transformasi penting untuk mempersiapkan data guna analisis lebih lanjut. Pertama, kolom tanggal dikonversi ke format datetime, sehingga memungkinkan ekstraksi informasi temporal seperti hari dalam seminggu dan bulan dari kolom 'order date (DateOrders)' serta 'shipping date (DateOrders)'. Informasi ini esensial untuk mengidentifikasi pola waktu dan tren musiman dalam data transaksi.

Selanjutnya, perhitungan shipping delay dilakukan dengan mengurangkan tanggal pengiriman dengan tanggal pesanan, dan nilai yang diperoleh dikategorikan menjadi

tiga kelas, yaitu 'on-time', 'slight delay', dan 'major delay', guna memberikan gambaran tentang efektivitas proses pengiriman. Selain itu, beberapa fitur baru dikonstruksi untuk meningkatkan daya prediktif data, antara lain Total_Discount yang dihitung sebagai perkalian antara total order dengan nilai diskon, Quantity_Discount yang merupakan hasil perkalian jumlah item dengan diskon, serta Distance yang dihitung menggunakan rumus Pythagoras berdasarkan nilai latitude dan longitude untuk mengestimasi jarak geografis.

Untuk mengurangi pengaruh nilai ekstrim pada target 'Order Profit Per Order', dilakukan penghilangan outlier dengan metode Interquartile Range (IQR), di mana data di luar batas Q1 – 1.5 IQR dan Q3 + 1.5 IQR dihapus dari dataset. Terakhir, dataset diurutkan berdasarkan 'Order Date' untuk memastikan bahwa data disusun secara kronologis, yang sangat penting dalam analisis berbasis waktu dan model prediksi.

d. Exploratory Data Analysis (EDA)

Untuk memahami distribusi dan hubungan antar variabel, dilakukan visualisasi fitur numerik menggunakan histogram dan box plot, serta fitur kategorikal menggunakan count plot. Analisis korelasi antar fitur numerik juga dilakukan dengan menggunakan heatmap, sehingga hubungan antar variabel yang mungkin mempengaruhi target dapat diidentifikasi secara jelas. Teknik-teknik ini membantu mendeteksi pola, outlier, dan potensi multikolinearitas yang perlu diatasi sebelum pembangunan model.

e. Pembagian Data Latih-Uji-Validasi

Strategi pembagian dataset dilakukan dengan mempertahankan urutan waktu. Data diurutkan berdasarkan tanggal pesanan dan kemudian dibagi menjadi tiga bagian: 60% sebagai data latih untuk membangun model, 20% sebagai data validasi untuk tuning model, dan 20% sebagai data uji untuk mengukur performa model pada data yang belum pernah dilihat sebelumnya. Pendekatan ini memastikan bahwa model dapat belajar dari data historis dan diuji pada kondisi yang mencerminkan tren terbaru.

f. Normalisasi

Pada tahap preprocessing, fitur numerik dinormalisasi menggunakan StandardScaler untuk menyamakan skala sehingga masing-masing fitur memiliki rata-rata nol dan standar deviasi satu. Hal ini penting agar fitur dengan skala besar tidak mendominasi proses pembelajaran. Sementara itu, fitur kategorikal di encoding menggunakan OneHotEncoder melalui Column Transformer, sehingga kategori dapat direpresentasikan dalam format numerik yang dapat diproses oleh algoritma pembelajaran mesin. Proses ini membantu model dalam mempercepat konvergensi selama pelatihan.

g. Pembuatan Urutan untuk Deret Waktu

Meskipun dataset ini tidak secara eksplisit merupakan data deret waktu, pengurutan data berdasarkan tanggal pesanan memberikan pendekatan serupa dengan prediksi deret waktu. Dengan menjaga urutan kronologis, model dapat menangkap dinamika perubahan data dari waktu ke waktu, sehingga mampu mendeteksi pola musiman dan tren jangka panjang yang penting untuk prediksi. Teknik ini sejalan dengan konsep pembuatan urutan data historis (misalnya, 24 jam terakhir) untuk memprediksi nilai di masa mendatang, yang sangat bermanfaat dalam analisis tren transaksi.

D. Modeling

a. Model LGBM Regressor

Model LGBMRegressor dibangun menggunakan pendekatan optimisasi hyperparameter berbasis Bayesian Optimization dengan Hyperopt. Proses dimulai dengan mendefinisikan fungsi objektif, di mana model diinisialisasi dengan serangkaian parameter yang mengatur kompleksitas struktur pohon, kecepatan pembelajaran, jumlah iterasi pelatihan, batas maksimum kedalaman pohon, ukuran minimum sampel pada setiap daun, serta proporsi data yang digunakan. Sebelum proses pelatihan, data training dan validasi dibersihkan dari nilai yang tidak valid untuk memastikan kestabilan perhitungan**.**

Selanjutnya, model dilatih dengan menerapkan mekanisme early stopping, yaitu pelatihan akan dihentikan secara otomatis apabila matrik evaluasi pada data validasi tidak menunjukkan peningkatan setelah sejumlah iterasi tertentu, sehingga dapat mencegah overfitting dan menghemat waktu komputasi. Setelah pelatihan, model menghasilkan prediksi pada data validasi dan kinerjanya diukur menggunakan suatu metrik yang dikonversi menjadi nilai yang diminimalkan oleh algoritma optimasi.

Ruang pencarian untuk parameter tersebut ditetapkan dalam rentang nilai tertentu, dan proses optimisasi dilakukan dengan algoritma estimasi probabilistik yang mengevaluasi beberapa kandidat untuk menemukan kombinasi parameter yang paling optimal. Setelah parameter terbaik diperoleh dan disesuaikan dengan tipe data yang diperlukan, model dilatih kembali pada data training dengan konfigurasi tersebut, tetap dilengkapi dengan mekanisme early stopping pada data validasi. Pendekatan ini mengintegrasikan pembersihan data, optimisasi hyperparameter, dan pencegahan overfitting secara menyeluruh, sehingga menghasilkan model prediktif yang handal dan efisien dalam menangani data yang kompleks dan berukuran besar.

b. Model XGBoost

Model XGBoost diimplementasikan sebagai metode berbasis pohon yang dirancang untuk menangkap hubungan non-linear di antara fitur-fitur dalam data rantai pasok dan logistik guna memprediksi profit per pesanan. Konfigurasi model ini mencakup pengaturan parameter yang mengontrol kompleksitas struktur pohon, kecepatan pembelajaran, serta jumlah iterasi pelatihan, sehingga fokus utamanya adalah meminimalkan kesalahan prediksi melalui metrik evaluasi yang sesuai.

Selain itu, model ini menerapkan mekanisme penghentian dini (early stopping) berdasarkan evaluasi performa pada data validasi, yang membantu menghentikan proses pelatihan ketika tidak terjadi peningkatan signifikan. Pendekatan ini memungkinkan XGBoost untuk menangani interaksi kompleks antar fitur dan sangat relevan dalam konteks keragaman produk, yang pada gilirannya mendukung identifikasi produk dengan margin keuntungan yang tinggi.

c. Model CatBoost

Model CatBoost dipilih untuk mengatasi tantangan pengolahan data yang mencakup fitur numerik dan kategorikal dalam konteks rantai pasok. Konfigurasi model ini dioptimalkan dengan mengatur parameter yang mempengaruhi laju pembelajaran, kedalaman model, dan teknik regulasi guna mencegah overfitting, sehingga model mampu menangkap pola non-linear yang sangat kompleks.

Selama proses pelatihan, evaluasi berkala pada data validasi diterapkan, dan mekanisme early stopping dimanfaatkan untuk menghentikan pelatihan saat tidak terjadi perbaikan performa dalam rentang iterasi tertentu. Ini memastikan bahwa CatBoost dapat mengelola data kategorik secara efisien dan meningkatkan akurasi prediksi profit per pesanan, terutama ketika dihadapkan pada beragam informasi transaksi, pengiriman, dan pesanan.

d. Model Stacking Ensemble

Dalam upaya meningkatkan akurasi prediksi secara keseluruhan, diterapkan teknik stacking ensemble yang mengintegrasikan tiga model base, yaitu LGBMRegressor, XGBoost, dan CatBoost. Pada pendekatan ini, masing-masing model base menghasilkan prediksi yang kemudian dikombinasikan melalui final estimator, yang pada implementasinya menggunakan model Elastic Net dengan parameter alpha dan l1 ratio. Penggunaan Elastic Net sebagai final estimator memungkinkan penggabungan regulasi L1 dan L2 untuk menangani potensi multikolinearitas di antara prediktor, sehingga memberikan stabilitas dan kekuatan dalam generalisasi model.

Proses pelatihan dilakukan dengan memanfaatkan data yang telah dipra-proses, di mana stacking ensemble secara efektif mengoptimalkan kekuatan masing-masing model base dengan mengurangi kelemahan individualnya dan meningkatkan kemampuan prediksi akhir terhadap Order Profit Per Order. Pendekatan ini, yang dilaksanakan melalui Stacking Regressor, memastikan bahwa prediksi dari ketiga model tersebut menghasilkan estimasi yang lebih robust dan akurat.

E. Pengujian dan Analisis

Pengujian dan evaluasi dilakukan untuk mengukur kinerja model prediksi Order Profit Per Order dengan menggunakan metrik evaluasi seperti Mean Absolute Error (MAE), Root Mean Square Error (RMSE), dan koefisien determinasi (R^2). Tujuan utama dari pengujian ini adalah menilai tingkat akurasi model dalam memprediksi keuntungan per pesanan berdasarkan data historis.

Analisis komparatif dilakukan untuk membandingkan berbagai pendekatan model, termasuk model regresi dan metode machine learning lainnya, guna mengidentifikasi strategi prediksi yang optimal. Evaluasi ini bertujuan untuk memahami keunggulan dan keterbatasan masing-masing model dalam menangani data serta memberikan hasil prediksi yang akurat. Dengan pendekatan ini, dapat dipilih model yang paling sesuai dengan kebutuhan operasional dan tujuan analisis.

F. Hasil dan Rekomendasi

a. Hasil Pengujian Model

LightGBM:

Dalam penerapan model LightGBM, data transaksi, pelanggan, dan logistik yang telah melalui proses pembersihan, transformasi, dan normalisasi digunakan sebagai dasar analisis. Dengan pengaturan hyperparameter yang dioptimalkan menggunakan teknik Bayesian Optimization, LightGBM berhasil menghasilkan MAE sebesar 12,39, RMSE sebesar 25,95, dan nilai R^2 mencapai 0,93. Performa ini menunjukkan bahwa model mampu menangkap pola non-linear, tren musiman, dan interaksi antar variabel dengan baik, sambil mempertahankan efisiensi komputasi yang tinggi, sehingga cocok untuk data berukuran besar dengan dinamika kompleks.

XGBoost:

Model XGBoost menggunakan mekanisme boosting untuk secara iteratif belajar dari kesalahan prediksi, sehingga mampu menangkap hubungan non-linear antar fitur yang ada dalam data. Dengan menggunakan parameter objective 'reg:squarederror' dan menerapkan early stopping pada proses training, XGBoost menghasilkan MAE sebesar 12,67, RMSE sebesar 27,18, serta R^2 sebesar 0,92. Hasil ini mengindikasikan bahwa meskipun terdapat sedikit perbedaan error

dibandingkan dengan model lain, XGBoost tetap efektif dalam mengoptimalkan prediksi melalui pembelajaran bertahap.

CatBoost:

CatBoost dipilih karena kemampuannya yang unggul dalam menangani data kategorikal secara otomatis, tanpa perlu banyak penyesuaian tambahan. Dengan konfigurasi seperti 800 iterasi, learning rate 0,05, dan kedalaman pohon 6, CatBoost berhasil mencapai MAE sebesar 11,96, RMSE sebesar 25,67, serta nilai R^2 sebesar 0,93. Hasil ini menunjukkan bahwa model ini efektif dalam mengelola interaksi antar variabel dan menangkap pola non-linear yang kompleks, sambil menjaga stabilitas dan mengurangi risiko overfitting.

Stacking Ensemble:

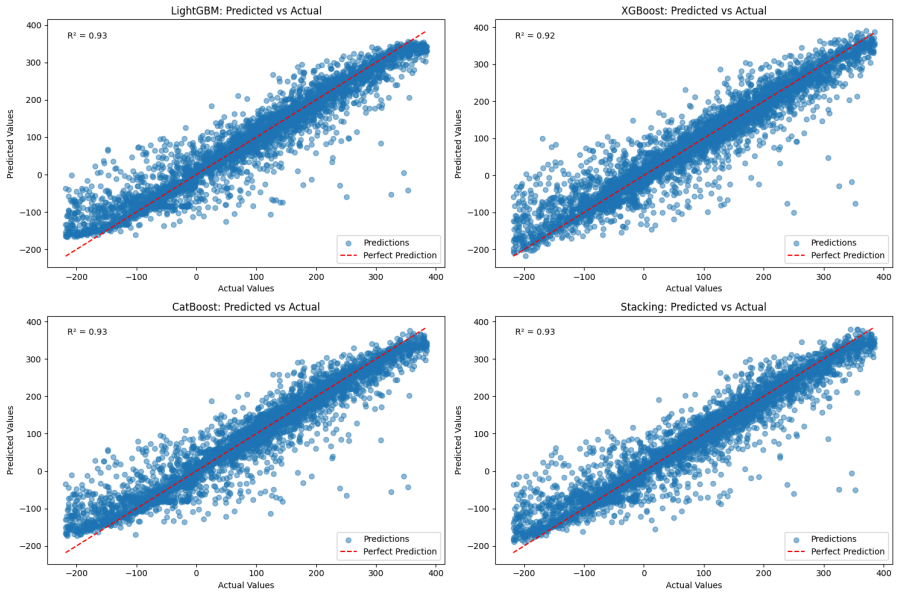
Pendekatan stacking ensemble menggabungkan keunggulan dari model LightGBM, XGBoost, dan CatBoost untuk menghasilkan prediksi yang lebih akurat. Dengan menggunakan ElasticNet sebagai final estimator, yang mengintegrasikan regulasi L1 dan L2 guna mengatasi potensi multikolinearitas, metode stacking berhasil menghasilkan MAE sebesar 11,38, RMSE sebesar 25,59, dan R^2 sebesar 0,93. Pendekatan ini memanfaatkan kelebihan masing-masing model dasar sehingga menghasilkan estimasi yang lebih konsisten dan robust, memberikan solusi yang optimal dalam menghadapi kerumitan data.

b. Perbandingan Model

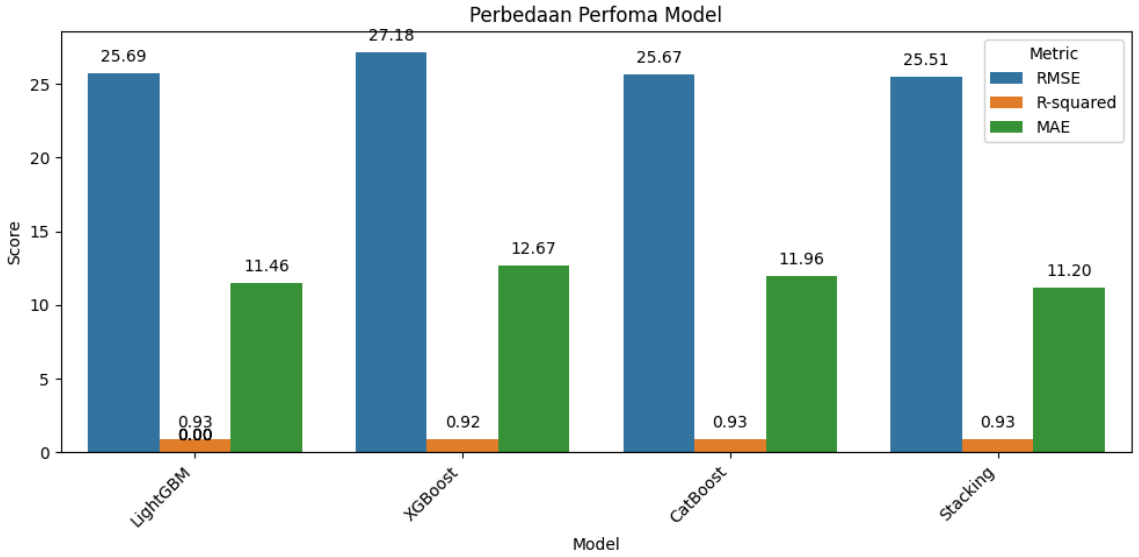
Keempat model menunjukkan kinerja yang cukup kompetitif dengan nilai R^2 mendekati 0,93, yang mengindikasikan bahwa masing-masing model mampu menjelaskan sekitar 93% variasi dalam data. Meskipun terdapat perbedaan yang relatif kecil pada metrik MAE dan RMSE, perbedaan tersebut mencerminkan keunggulan spesifik dari setiap model. Berikut adalah visualisasi perbandingan setiap model :

Model	MAE	RMSE	R^2
LightGBM	12,39	25,95	0,93
XGBoost	12,67	27,18	0,92
CatBoost	11,96	25,67	0,93
Stacking Ensemble	11,38	25,59	0,93

(Tabel Perbandingan Evaluasi Model)



(Visualisasi Nilai Prediktif dan Aktual)



(Visualisasi Perbandingan Performa Evaluasi Model)

c. Kesimpulan

Berdasarkan Hasil analisis, data yang telah diproses melalui serangkaian tahapan pembersihan, agregasi, dan feature engineering, model-model berbasis pohon menunjukkan keunggulan dalam menangkap dinamika data yang kompleks. LightGBM Regressor secara efisien mampu mengidentifikasi pola non-linear dan tren musiman dengan kecepatan tinggi, sehingga memberikan fondasi yang kuat untuk prediksi. Sebelumnya, percobaan terhadap model linear seperti regresi linier dan model-model sejenis hanya mampu menghasilkan performa dengan tingkat akurasi di bawah 40%, yang menandakan keterbatasan pendekatan linear dalam mengatasi pola non-linear dan interaksi kompleks antar fitur.

XGBoost Regressor, meskipun telah dioptimalkan menggunakan teknik Bayesian Optimization untuk pengaturan hyperparameter, tetap menghasilkan prediksi yang secara kualitatif kalah akurasi dibandingkan model lain yang kemungkinan mekanisme early stopping yang diterapkan belum sepenuhnya mampu mengatasi kompleksitas interaksi antar fitur. Di sisi lain, CatBoost Regressor unggul dalam pengolahan fitur kategorikal dan mampu mendeteksi interaksi non-linear secara mendalam, sehingga memberikan prediksi yang lebih akurat dalam konteks kerumitan data.

Pendekatan stacking ensemble, yang mengintegrasikan kelebihan dari ketiga model tersebut melalui final estimator berbasis ElasticNet, menghasilkan prediksi yang paling konsisten dan robust karena menggabungkan kekuatan masing-masing model dasar. Secara keseluruhan, temuan ini mendukung rekomendasi bahwa untuk aplikasi dengan data kompleks dan dinamis, model berbasis pohon khususnya melalui pendekatan ensemble merupakan solusi optimal, sedangkan model linear tetap relevan ketika interpretabilitas menjadi prioritas utama.

d. Rekomendasi

Berdasarkan analisis hasil model, disarankan untuk mengembangkan pendekatan hibrida yang mengkombinasikan kelebihan model regresi linier dengan model berbasis pohon. Pendekatan ini memungkinkan pemanfaatan aspek interpretabilitas dari model linier serta kemampuan menangkap pola kompleks dan non-linear dari model pohon. Meskipun model regresi linier menunjukkan performa yang kurang maksimal dibandingkan dengan model berbasis pohon, keberadaannya sebagai baseline tetap penting untuk percobaan dan evaluasi komparatif.

Selain itu, peningkatan resolusi data dengan menggunakan data yang lebih granular serta penambahan variabel eksternal seperti kondisi cuaca, kalender kegiatan, atau indikator ekonomi perlu dipertimbangkan guna memperoleh wawasan yang lebih mendetail mengenai dinamika sistem.

Selanjutnya, disarankan untuk mengadopsi metode validasi deret waktu yang lebih optimal, seperti modifikasi k-fold cross-validation untuk data serial, agar model dapat diuji secara lebih mendalam terhadap kemampuan generalisasinya. Optimasi lebih lanjut pada proses pemilihan fitur melalui analisis pentingnya fitur (feature importance analysis) juga akan membantu model untuk lebih fokus pada variabel-variabel kunci yang berpengaruh signifikan terhadap prediksi.