

Prediksi Pengiriman Bermasalah PT Lion Express Palembang: Perbandingan Tiga Model Klasifikasi

Muktiono^{1*}, Rusmala Santi²

Program Studi Sistem Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Raden Fatah,
Palembang, Indonesia^{1,2}

*Email Korespondensi: mukti290804@gmail.com

Sejarah Artikel:

Diterima 17-10-2026
Disetujui 23-10-2026
Diterbitkan 25-10-2026

ABSTRACT

Delivery exceptions raise handling costs and reduce customer satisfaction, yet their small share makes manual identification difficult. This study compares Logistic Regression, Decision Tree, and Random Forest in predicting deliveries not yet marked as proof of delivery (non-POD) at PT Lion Express Palembang Branch, using destination area, chargeable weight, delivery cycle, and day of delivery, and evaluates class imbalance handling. The data comprise 32,475 transactions from August 2026, of which 8.52% are problematic. Data were split by waybill number to prevent leakage, hyperparameters were tuned with 5-fold cross-validation using PR-AUC, and class imbalance was handled with SMOTENC. On the test set, Random Forest achieved a ROC-AUC of 0.745 (95% CI 0.719–0.770) and a PR-AUC of 0.370 (4.3 times the random baseline of 0.085); its difference from Decision Tree was not significant. SMOTENC did not improve ROC-AUC or PR-AUC but shifted the operating point, while tuning raised PR-AUC by about 0.03. Destination area was the most important feature, followed by day, cycle, and chargeable weight. Under temporal validation, ROC-AUC fell to 0.666 and PR-AUC to 0.143, so the model suits monitoring prioritisation better than automated decisions.

Keywords: *classification; machine learning; delivery exception; imbalanced data; SMOTENC; Random Forest*

ABSTRAK

Pengiriman bermasalah (delivery exception) menambah biaya penanganan dan menurunkan kepuasan pelanggan, tetapi proporsinya yang kecil menyulitkan identifikasi manual. Penelitian ini membandingkan Logistic Regression, Decision Tree, dan Random Forest untuk memprediksi pengiriman yang belum berstatus bukti pengiriman (non-POD) pada PT Lion Express Cabang Palembang berdasarkan area tujuan, berat tertagih, siklus, dan hari pengiriman, serta mengevaluasi penanganan ketidakseimbangan kelas. Data terdiri atas 32.475 transaksi Agustus 2026 dengan proporsi kelas bermasalah 8,52%. Data dibagi per nomor resi untuk mencegah kebocoran, hyperparameter dioptimasi dengan validasi silang 5-lipatan ber kriteria PR-AUC, dan ketidakseimbangan kelas ditangani dengan SMOTENC. Pada data uji, Random Forest memperoleh ROC-AUC 0,745 (CI 95%: 0,719–0,770) dan PR-AUC 0,370 (4,3 kali nilai acak 0,085); selisihnya dengan Decision Tree tidak signifikan. SMOTENC tidak meningkatkan ROC-AUC maupun PR-AUC, tetapi menggeser titik operasi, sedangkan tuning menaikkan PR-AUC sekitar 0,03. Area tujuan merupakan fitur terpenting, diikuti hari, siklus, dan berat tertagih. Pada validasi temporal, ROC-AUC turun menjadi 0,666 dan PR-AUC 0,143, sehingga model lebih sesuai untuk prioritas pengawasan daripada keputusan otomatis.

Katakunci: *klasifikasi; machine learning; pengiriman bermasalah; data tidak seimbang; SMOTENC; Random Forest*

Bagaimana Cara Sitasi Artikel ini:

Muktiono, M., & Santi, R. . (2026). Prediksi Pengiriman Bermasalah PT Lion Express Palembang: Perbandingan Tiga Model Klasifikasi. Jejak Digital: Jurnal Ilmiah Multidisiplin, 2(6), 11088-11100. <https://doi.org/10.63822/q3ntna05>

PENDAHULUAN

Kegagalan pengiriman merupakan masalah operasional yang nyata di industri ekspedisi. Lim dkk. (2023) melaporkan bahwa hingga 20% pengiriman business-to-consumer gagal pada percobaan pertama. Pengiriman bermasalah (*delivery exception*) menambah biaya penanganan ulang dan berpotensi menurunkan kepuasan pelanggan. PT Lion Express Cabang Palembang mencatat lebih dari 32.000 transaksi pada Agustus 2026, dan sekitar 8,5% di antaranya berstatus bukan POD (*proof of delivery*). Pada volume sebesar ini, mengenali pengiriman berisiko secara manual tidak praktis.

Pendekatan *machine learning* telah dipakai untuk memprediksi hasil pengiriman. Lim dkk. (2023) memanfaatkan prediktor percobaan pengantaran rumah dari data peritel daring di Amerika Selatan untuk memperbaiki penentuan rute, dan de Leeuw dkk. (2023) membangun model prediksi kehilangan paket pada data setahun pengiriman di Belgia dengan menyeimbangkan data sebelum pembelajaran terawasi. Kedua studi tersebut menggunakan data dan konteks operasional yang berbeda dari cabang ekspedisi di Indonesia. Pada literatur berbahasa Indonesia, Pambudi dkk. (2020) membandingkan regresi logistik, jaringan saraf tiruan, dan Random Forest untuk memperkirakan status pengiriman barang pada perusahaan *freight forwarder* (kargo), dengan Random Forest terbaik (akurasi 76,6%). Studi tersebut melaporkan akurasi sebagai ukuran kinerja, padahal pada data yang tidak seimbang akurasi dapat menyesatkan. Pada penelusuran terbatas yang dilakukan penulis, belum ditemukan studi yang mengevaluasi prediksi pengiriman bermasalah pada data operasional cabang ekspedisi di Indonesia dengan metrik yang sesuai untuk data tidak seimbang.

Tantangan utama adalah ketidakseimbangan kelas: pengiriman bermasalah hanya sekitar 8,5%. SMOTE (Chawla dkk., 2002) lazim dipakai untuk mengatasinya, dan beberapa studi berbahasa Indonesia melaporkan manfaatnya pada Random Forest (Erlin dkk., 2022; Hayati dkk., 2024). Namun bukti terbaru lebih berhati-hati. Elor dan Averbuch-Elor (2022) menemukan bahwa penyeimbangan data tidak meningkatkan kinerja klasifikator yang kuat, dan van den Goorbergh dkk. (2022) menunjukkan bahwa koreksi ketidakseimbangan tidak memperbaiki AUROC serta menyebabkan estimasi probabilitas kelas minoritas terlalu tinggi. Karena itu manfaat SMOTE dan *hyperparameter tuning* perlu diuji secara empiris, bukan diasumsikan.

Penelitian ini bertujuan (1) membandingkan Logistic Regression, Decision Tree, dan Random Forest untuk memprediksi pengiriman bermasalah; (2) mengevaluasi penanganan ketidakseimbangan kelas (SMOTENC, SMOTE standar, dan *class_weight*) serta kontribusi *hyperparameter tuning* melalui analisis ablasi; (3) menilai kontribusi tiap fitur dengan *permutation importance*; dan (4) menilai kelayakan operasional model, termasuk ketahanannya pada validasi temporal. Kontribusi penelitian adalah evaluasi dengan interval kepercayaan, pembagian data per nomor resi, dan pembahasan beban operasional akibat peringatan keliru.

METODE PENELITIAN

Data dan Pembersihan

Data berupa 32.859 baris transaksi pengiriman PT Lion Express Cabang Palembang periode 1–31 Agustus 2026 dengan variabel tanggal, siklus (CYCLE), nomor resi (STT NO), area tujuan (SHUTTLE), berat tertagih (CW), status akhir (LAST STATUS), dan kurir. Pembersihan data meliputi: (1) penghapusan

372 baris duplikat persis; (2) penghapusan 12 baris tanpa nilai CW; (3) penyeragaman format tanggal (sebagian tercatat sebagai teks, mis. “14-Agu-26”); dan (4) pembacaan nilai CW berformat desimal koma (427 baris bernilai 0,1–0,3). Hasil akhirnya adalah 32.475 transaksi dengan 31.415 nomor resi unik; 964 nomor resi muncul lebih dari satu kali (mis. pengiriman ulang). Data diperoleh melalui kegiatan magang penulis di PT Lion Express Cabang Palembang. Variabel kurir tidak digunakan dalam pemodelan.

Variabel target ditetapkan sebagai Bermasalah bila status akhir bukan POD, dan Berhasil bila POD. Diperoleh 29.708 transaksi Berhasil (91,48%) dan 2.767 Bermasalah (8,52%). Status Bermasalah terdiri atas DEX (1.973; 71,3%), KONDISPATCH (604; 21,8%), HAL (90; 3,3%), STLDISPATCH (80; 2,9%), dan 20 transaksi berstatus lain (0,7%). Menurut keterangan pembimbing lapangan, DEX adalah status ketika barang tidak dapat dikirim karena alasan tertentu; STLDISPATCH adalah status ketika penugasan kurir atau armada pengirim telah dikonfirmasi di sistem operasional perusahaan sehingga paket siap atau sedang dibawa menuju titik distribusi berikutnya; sedangkan KONDISPATCH dan HAL, yang dijelaskan bersama, menandakan konfirmasi atau instruksi penugasan kurir atau armada di titik transit untuk meneruskan paket. Dengan demikian, hanya DEX yang secara tegas menunjukkan pengiriman tidak berhasil, sedangkan STLDISPATCH dan KONDISPATCH dapat menandai paket yang masih dalam proses. Kategori Bermasalah dalam penelitian ini karena itu diartikan sebagai pengiriman yang belum berstatus POD pada saat data diambil. CW (chargeable weight) adalah berat tertagih yang dinyatakan dalam kilogram dan SHUTTLE adalah area tujuan pengiriman, sedangkan siklus (CYCLE) C.1–C.4, menurut keterangan yang sama, adalah perulangan atau putaran proses pengiriman. Kelas Bermasalah diperlakukan sebagai kelas positif pada seluruh metrik.

Dataset memuat 104 area tujuan dengan sebaran sangat timpang: 17 area memiliki sedikitnya 30 pengiriman, sedangkan 80 area kurang dari 10 pengiriman. Berat tertagih bermedian 1 kg (rentang 0,1–400 kg).

Rekayasa Fitur

Empat informasi dasar diolah menjadi 13 kolom masukan (Tabel 1). Seluruh *encoding* dan penskalaan dipelajari hanya dari data latih (di dalam setiap lipatan validasi silang) dan diterapkan ke data validasi/uji.

Tabel 1. Fitur Prediktor dan Perlakuan Encoding

Fitur	Jenis data	Perlakuan
SHUTTLE_FRE Q	Numerik	Area tujuan (104 kategori) diubah menjadi frekuensi kemunculan relatif di data latih (frequency encoding); area yang tidak ada di data latih bernilai 0. Fitur ini mewakili volume relatif area.
CW	Numerik	Berat tertagih ditransformasi $\log(1 + CW)$ karena sangat condong ke kanan, lalu dibakukan (StandardScaler).
DAY_OF_WEEK	Kategorikal	Hari pengiriman, one-hot encoding (7 kolom), tanpa urutan semu.
CYCLE	Kategorikal	Siklus C.1–C.4, one-hot encoding (4 kolom).

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Frequency encoding menghindari urutan semu *label encoding* tanpa menambah ratusan kolom, tetapi dua area berbeda dengan frekuensi sama menjadi tidak terbedakan. Sebagai pembanding diuji *target encoding* yang dihitung secara *out-of-fold* (Micci-Barreca, 2001), serta penyandian hari sebagai angka 0–6

ditambah penanda akhir pekan (Subbab 3.2).

Penanganan Ketidakseimbangan Kelas

Rasio kelas sekitar 1:11. Digunakan SMOTENC, varian SMOTE (Chawla dkk., 2002) untuk fitur campuran numerik-kategorikal: fitur numerik diinterpolasi antar-tetangga ($k = 5$), sedangkan fitur kategorikal (hari dan siklus) diambil dari modulus tetangga sehingga kolom *one-hot* tetap bernilai 0/1. Kelas minoritas dibangkitkan hingga seimbang 1:1. Oversampling hanya dilakukan pada bagian latih di setiap lipatan, sehingga data validasi dan uji tidak pernah berisi data sintetis. Pustaka imbalanced-learn (Lemaître dkk., 2017) tidak tersedia pada lingkungan komputasi penelitian ini, sehingga SMOTENC diimplementasikan sendiri dengan scikit-learn (Pedregosa dkk., 2011) mengikuti algoritma aslinya. SMOTE standar, *class_weight*, dan tanpa penyeimbangan dibandingkan dalam analisis ablasi. SMOTENC dipertahankan sebagai konfigurasi utama bukan karena meningkatkan metrik yang bebas ambang (Subbab 3.2), melainkan karena pada ambang bawaan 0,5 model tanpa penyeimbangan hampir tidak menandai pengiriman bermasalah (recall 0,128), sehingga kurang berguna untuk pengawasan.

Algoritma dan Optimasi Hyperparameter

Tiga algoritma dibandingkan: Logistic Regression, Decision Tree, dan Random Forest (Breiman, 2001). Hyperparameter dioptimasi dengan pencarian grid dan validasi silang 5-lipatan yang tergrup per nomor resi (StratifiedGroupKFold) dengan kriteria PR-AUC pada kelas Bermasalah, yaitu metrik yang bebas dari ambang. Ruang pencarian dan hasilnya disajikan pada Tabel 2.

Tabel 2. Ruang Pencarian dan Hyperparameter Terbaik (Model dengan SMOTENC)

Model	Parameter yang diuji	Terbaik	PR-AUC CV
Logistic Regression	C: 0,001; 0,01; 0,1; 1; 10	C = 0,001 (batas bawah grid)	0,187
Decision Tree	max_depth: 2, 3, 4, 6, 8, 10, None; min_samples_leaf: 1, 5, 10, 20, 50	max_depth = None; min_samples_leaf = 50 (batas atas grid)	0,316
Random Forest	n_estimators: 100, 300; max_depth: 6, 10, 14, None; min_samples_leaf: 1, 5, 10, 20, 50	n_estimators = 300; max_depth = 10; min_samples_leaf = 20	0,335

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Nilai terbaik Logistic Regression dan Decision Tree berada di tepi grid, sehingga optimum sesungguhnya mungkin lebih ekstrem; hal ini dicatat sebagai keterbatasan. Seluruh pengaturan Logistic Regression menghasilkan PR-AUC validasi yang berdekatan, yang menunjukkan bahwa model linear kurang cocok untuk fitur ini.

Desain Evaluasi

Data dibagi 80:20 secara berstrata dan tergrup per nomor resi, sehingga tidak ada nomor resi yang muncul di data latih sekaligus data uji (25.980 latih; 6.495 uji, di antaranya 553 Bermasalah atau 8,51%). Metrik pada kelas Bermasalah: precision, recall, dan F1-score pada ambang 0,5, serta ROC-AUC dan PR-

AUC. Kurva ROC tidak berubah oleh distribusi kelas (Fawcett, 2006), tetapi PR-AUC bergantung pada prevalensi, sehingga ia selalu dibaca terhadap nilai acaknya (sama dengan prevalensi, 0,085) dan prevalensi dilaporkan bersama hasil (Davis & Goadrich, 2006; Saito & Rehmsmeier, 2015; Richardson dkk., 2024). Akurasi hanya dilaporkan sebagai pelengkap karena menyesatkan pada data tidak seimbang.

Ketidakpastian diukur dengan *bootstrap* 1.000 kali yang mengambil sampel per nomor resi (interval kepercayaan 95%), dan perbandingan antar-model memakai selisih berpasangan pada sampel *bootstrap* yang sama. Kontribusi fitur dinilai dengan *permutation importance* pada data uji (15 pengulangan; Breiman, 2001), dibandingkan dengan importance berbasis penurunan impuritas, yang dikenal bias terhadap fitur dengan banyak nilai unik (Strobl dkk., 2007). Untuk menguji generalisasi ke waktu, dilakukan validasi temporal: latih pada 1–21 Agustus (22.413 transaksi) dan uji pada 22–31 Agustus (10.006 transaksi) setelah membuang nomor resi yang muncul di kedua periode. Percobaan dijalankan dengan scikit-learn 1.8.0 (*random state* pembagian 42, model 0, *bootstrap* 2026).

HASIL DAN PEMBAHASAN

Gambaran Data dan Perbandingan Model

Proporsi pengiriman bermasalah bervariasi menurut hari, siklus, dan area (Tabel 3 dan 4). Hari Minggu paling tinggi (17,3%), dan kelima hari Minggu pada Agustus semuanya di atas rata-rata (11,5–29,7%). Siklus C.4 (13,9%) dan C.2 (11,3%) lebih tinggi daripada C.1 (7,0%). Berat tertagih berhubungan lemah dengan proporsi masalah (6,8–10,7% pada semua kelompok berat tertagih). Terdapat pula lonjakan harian, misalnya 29,7% pada 9 Agustus dan 31,4% pada 11 Agustus. Pada area dengan sedikitnya 30 pengiriman (17 area), volume berkorelasi negatif dengan proporsi masalah (Spearman $\rho = -0,57$; $p = 0,018$): area bervolume kecil dan umumnya di luar kota, seperti Indralaya (37,7%) dan Tanjung Raja (28,2%), lebih sering bermasalah.

Tabel 3. Proporsi Pengiriman Bermasalah menurut Hari dan Siklus

Kelompok	Jumlah	Bermasalah (%)	Kelompok	Jumlah	Bermasalah (%)
Senin	4.756	7,3	C.1	20.737	7,0
Selasa	3.033	9,8	C.2	3.713	11,3
Rabu	3.377	5,8	C.3	3.554	7,8
Kamis	4.691	6,7	C.4	4.471	13,9
Jumat	5.541	6,4			
Sabtu	5.785	5,9			
Minggu	5.292	17,3			

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Tabel 4. Proporsi Pengiriman Bermasalah pada Beberapa Area

Area	Jumlah	Bermasalah (%)	Area	Jumlah	Bermasalah (%)
Sako	5.130	3,3	Kertapati	1.176	12,9
Kalidoni	5.248	5,8	Prabumulih Barat	577	17,7
Sukarami	5.595	6,8	Tanjung Raja	117	28,2
Gandus	4.298	8,1	Indralaya	308	37,7
Plaju	3.065	9,0	08AL-16	149	39,6
Bukit Kecil	5.592	10,0	08SR-01	177	46,3

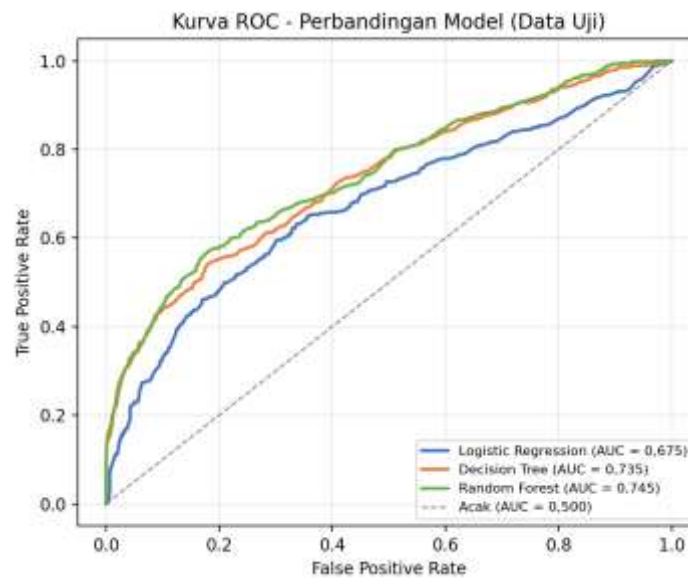
(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Tabel 5 menyajikan kinerja model pada data uji. Random Forest memperoleh ROC-AUC dan PR-AUC tertinggi, dengan PR-AUC sekitar 4,3 kali nilai acak (0,370 berbanding 0,085), disusul Decision Tree (4,2 kali) dan Logistic Regression (2,4 kali). Akurasi masing-masing 81,0%, 75,1%, dan 67,6% (Random Forest, Decision Tree, Logistic Regression). Selisih Random Forest terhadap Decision Tree pada ROC-AUC (+0,010; CI 95% -0,003 hingga 0,024) dan PR-AUC (+0,010; -0,006 hingga 0,025) tidak signifikan, sehingga keduanya setara dalam kemampuan mengurutkan risiko. Random Forest unggul jelas atas Logistic Regression (ROC-AUC +0,070; PR-AUC +0,166; kedua CI tidak memuat nol). Pada F1-score di ambang 0,5, Random Forest lebih tinggi daripada Decision Tree (+0,052; 0,039–0,065), tetapi selisih ini bergantung pada ambang yang dipilih. ROC-AUC 0,745 tergolong moderat.

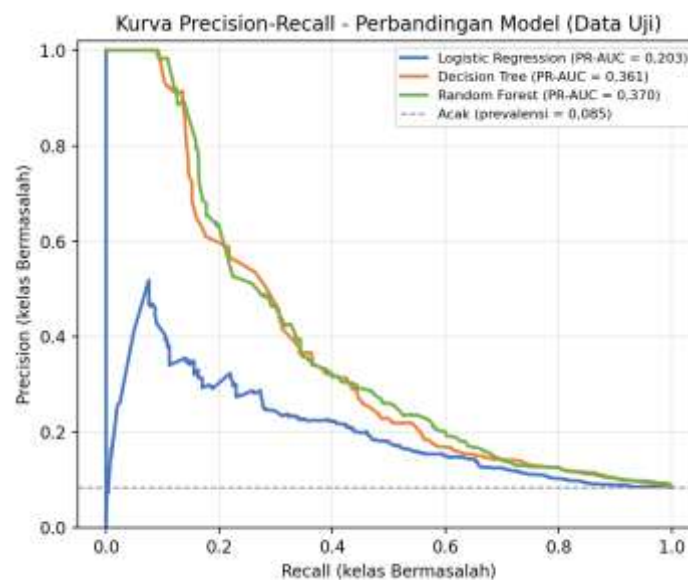
Tabel 5. Kinerja Model pada Data Uji (Kelas Positif: Bermasalah; CI 95% bootstrap)

Model	Precision	Recall	F1-score	ROC-AUC (CI 95%)	PR-AUC (CI 95%)
Logistic Regression	0,150	0,602	0,240	0,675 (0,646–0,702)	0,203 (0,176–0,235)
Decision Tree	0,185	0,564	0,278	0,735 (0,709–0,760)	0,361 (0,322–0,402)
Random Forest	0,236	0,550	0,330	0,745 (0,719–0,770)	0,370 (0,329–0,411)
Acuan acak	0,085	–	–	0,500	0,085

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)



Gambar 1. Kurva ROC Ketiga Model pada Data Uji
(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)



Gambar 2. Kurva Precision–Recall Ketiga Model pada Data Uji (garis putus-putus: nilai acak)
(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Pada Gambar 1, kurva Random Forest dan Decision Tree hampir berimpit di sebagian besar rentang, sedangkan Logistic Regression berada di bawah keduanya. Gambar 2 menunjukkan bahwa precision tinggi hanya dapat dicapai pada recall rendah: precision turun dari sekitar 0,6 pada recall 0,2 menjadi di bawah 0,2 pada recall 0,6.

Evaluasi Penanganan Ketidakseimbangan Kelas dan Analisis Ablasi

Tabel 6 membandingkan skenario pada Random Forest. *Hyperparameter tuning* meningkatkan PR-

AUC: tanpa SMOTENC dari 0,359 menjadi 0,385 (+0,026; CI 95% 0,012–0,040). Sebaliknya, SMOTENC tidak meningkatkan metrik yang tidak bergantung pada ambang. Dibandingkan model tuning tanpa SMOTENC, ROC-AUC tidak berbeda (–0,004; –0,012 hingga 0,002) dan PR-AUC sedikit lebih rendah (–0,014; –0,024 hingga –0,004).

Tabel 6. Hasil Ablasi Random Forest pada Data Uji (ambang 0,5)

Skenario	Precision	Recall	F1	ROC-AUC	PR-AUC	Ditandai
Acuan acak	–	–	–	0,500	0,085	–
(a) Bawaan, tanpa SMOTENC	0,691	0,154	0,251	0,736	0,359	123
(b) Bawaan + <i>class_weight</i>	0,196	0,523	0,285	0,723	0,352	1.478
(c) Bawaan + SMOTENC	0,191	0,517	0,279	0,717	0,346	1.494
(d) Tuning, tanpa SMOTENC	0,947	0,128	0,226	0,749	0,385	75
(e) Tuning + SMOTENC (model utama)	0,236	0,550	0,330	0,745	0,370	1.289
(f) Tuning + <i>class_weight</i>	0,229	0,553	0,323	0,747	0,375	1.339
(g) Tuning + SMOTE standar	0,217	0,573	0,314	0,743	0,369	1.464

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Efek SMOTENC terlihat pada titik operasi. Tanpa penyeimbangan, model pada ambang 0,5 hanya menandai 75 pengiriman (1,2% data uji) dengan recall 0,128, sedangkan dengan SMOTENC model menandai 1.289 pengiriman dan F1-score naik +0,104 (0,054–0,152). Hasil serupa diperoleh dengan *class_weight* (skenario f; F1 0,323), sehingga SMOTENC tidak memberi keunggulan atas cara yang lebih sederhana. SMOTENC dan SMOTE standar juga tidak berbeda pada PR-AUC (+0,001; –0,004 hingga 0,007), meskipun SMOTENC memberi F1 yang sedikit lebih baik (+0,016). Temuan ini sejalan dengan Elor dan Averbuch-Elor (2022) serta van den Goorbergh dkk. (2022): penyeimbangan terutama menggeser ambang keputusan, bukan memperbaiki kemampuan membedakan kelas. Karena itu klaim bahwa SMOTE “meningkatkan kinerja” (Erllin dkk., 2022; Hayati dkk., 2024) perlu dibaca menurut metrik yang dipakai: pada F1 di ambang bawaan SMOTE tampak membantu, tetapi pada ROC-AUC dan PR-AUC tidak.

Dua variasi encoding tidak memperbaiki hasil. *Target encoding* untuk area menurunkan PR-AUC (0,358; selisih –0,012 dengan CI –0,022 hingga –0,003) dan F1 (0,282), sedangkan penyandian hari sebagai angka 0–6 ditambah penanda akhir pekan memberikan PR-AUC yang sama (0,370). Karena itu frequency encoding dan *one-hot* hari dipertahankan.

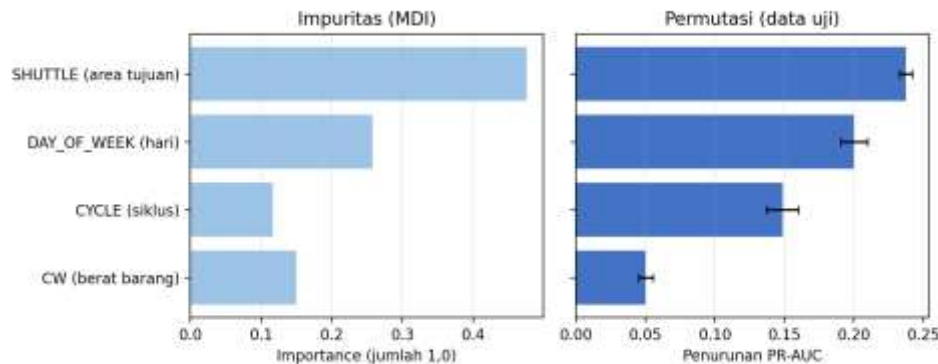
Kontribusi Fitur

Tabel 7 dan Gambar 3 memperlihatkan kontribusi fitur. Berdasarkan *permutation importance*, mengacak area tujuan menurunkan PR-AUC paling besar (0,237), diikuti hari (0,200), siklus (0,149), dan berat tertagih (0,050). Urutan ini berbeda dari importance berbasis impuritas, yang menempatkan berat tertagih (0,150) di atas siklus (0,117). Perbedaan tersebut sejalan dengan bias impuritas terhadap fitur numerik kontinu (Strobl dkk., 2007) dan sejalan pula dengan analisis deskriptif: berat tertagih berhubungan lemah dengan proporsi masalah (Subbab 3.1).

Tabel 7. Kontribusi Fitur pada Random Forest (Model Utama)

Fitur	Impuritas (MDI)	Penurunan PR-AUC (permutasi, $\pm$ SD)	Penurunan ROC-AUC (permutasi)
Area tujuan (SHUTTLE_FREQ)	0,476	0,237 $\pm$ 0,005	0,150
Hari (DAY_OF_WEEK)	0,258	0,200 $\pm$ 0,010	0,098
Siklus (CYCLE)	0,117	0,149 $\pm$ 0,011	0,054
Berat tertagih (CW)	0,150	0,050 $\pm$ 0,005	0,030

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)



Gambar 3. Importance Fitur Random Forest: Impuritas (kiri) dan Permutasi pada Data Uji (kanan)

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Fitur area bernilai tinggi karena SHUTTLE_FREQ mewakili volume relatif area, dan area bervolume kecil (umumnya di luar kota) lebih sering bermasalah (Subbab 3.1). Interpretasinya adalah asosiasi, bukan sebab-akibat. Mengingat 80 dari 104 area hanya memiliki kurang dari 10 pengiriman, fitur ini pada dasarnya juga bertindak sebagai pengenalan area langka. Pengaruh hari perlu dibaca bersama kenyataan bahwa data hanya satu bulan: setiap hari dalam sepekan hanya diwakili empat atau lima tanggal, sehingga fitur hari dapat menyerap kejadian pada tanggal tertentu (mis. lonjakan 9–13 Agustus).

Validasi Temporal

Pembagian acak menempatkan hari yang sama pada data latih dan uji, sehingga dapat terlalu optimistis. Pada validasi temporal (Tabel 8), kinerja menurun tajam: ROC-AUC Random Forest turun dari 0,745 menjadi 0,666 dan PR-AUC dari 0,370 menjadi 0,143, atau sekitar 1,8 kali nilai acak (prevalensi periode uji 7,7%). Random Forest tetap memiliki nilai numerik tertinggi, dengan selisih yang tipis terhadap Logistic Regression dan Decision Tree. Model tanpa penyeimbangan hanya menandai 27 dari 10.006 pengiriman (recall 0,005) pada ambang 0,5, menegaskan bahwa ambang bawaan tidak cocok tanpa penyeimbangan atau penyesuaian ambang.

Tabel 8. Validasi Temporal (Latih 1–21 Agustus, Uji 22–31 Agustus)

Model	ROC-AUC	PR-AUC	Precision	Recall	F1
Logistic Regression	0,630	0,135	0,132	0,459	0,205
Decision Tree	0,640	0,125	0,132	0,367	0,194
Random Forest	0,666	0,143	0,162	0,355	0,223
Random Forest tanpa SMOTENC	0,660	0,139	0,148	0,005	0,010
Acuan acak	0,500	0,077	–	–	–

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Implikasi Operasional

Pada ambang 0,5 dan data uji acak, Random Forest menghasilkan 304 positif benar, 985 positif keliru, 249 negatif keliru, dan 4.957 negatif benar. Model menandai 1.289 pengiriman (19,8% data uji) untuk menangkap 55,0% kasus bermasalah; precision 0,236 berarti sekitar 76% tanda adalah peringatan keliru, meskipun masih 2,8 kali lebih baik daripada memilih acak. Beban tindak lanjut sebesar itu kemungkinan tidak realistis, sehingga model lebih sesuai dipakai sebagai pemeringkat risiko (Tabel 9): 5% pengiriman berskor tertinggi memuat 48,6% kasus bermasalah (5,7 kali nilai acak) dan menangkap 28,6% dari seluruh kasus, sedangkan 20% teratas menangkap 55,2% kasus. Menaikkan ambang ke 0,7 memberikan F1 tertinggi (0,365) dengan hanya 7,1% pengiriman ditandai. Karena SMOTENC menggeser estimasi probabilitas ke atas (van den Goorbergh dkk., 2022), skor model sebaiknya dibaca sebagai peringkat, bukan probabilitas sebenarnya. Pada validasi temporal peringkat ini melemah (Subbab 3.4), sehingga model perlu dilatih ulang secara berkala. Dalam kerangka sistem informasi cabang, skor risiko dapat ditampilkan pada *dashboard* peringatan dini per resi dan per area.

Tabel 9. Efek Ambang dan Pemeringkatan pada Random Forest (Data Uji)

Ambang	Precision	Recall	F1	% ditandai	Top-k	Precision	Recall
0,3	0,100	0,924	0,180	78,9	5%	0,486	0,286
0,4	0,147	0,693	0,243	40,1	10%	0,331	0,389
0,5	0,236	0,550	0,330	19,8	20%	0,235	0,552
0,6	0,304	0,436	0,358	12,2	30%	0,180	0,635
0,7	0,402	0,335	0,365	7,1			
0,8	0,691	0,170	0,273	2,1			

(Sumber: Data PT Lion Express Cabang Palembang, diolah penulis, 2026)

Keterbatasan

Pertama, data hanya satu bulan dari satu cabang, dan penurunan kinerja pada validasi temporal menunjukkan bahwa hasil pembagian acak terlalu optimistis; data multi-bulan diperlukan. Kedua, label berasal dari status akhir saat data diambil; status seperti KONDISPATCH dan STLDISPATCH (24,7% kasus bermasalah) menurut keterangan pembimbing lapangan merupakan status dalam proses penugasan dan pengiriman, bukan kegagalan. Kategori Bermasalah karenanya mencakup pengiriman yang belum berstatus POD, bukan hanya yang gagal terkirim, sehingga hasil sensitif terhadap definisi label; penelitian lanjutan sebaiknya memodelkan DEX saja sebagai target. Ketiga, hanya empat fitur yang digunakan; informasi seperti jarak, kurir, atau riwayat area belum dimanfaatkan. Keempat, fitur area bersifat timpang

(80 area kurang dari 10 pengiriman), dan frequency encoding tidak membedakan area berfrekuensi sama. Kelima, hyperparameter Logistic Regression dan Decision Tree berada di tepi grid. Keenam, nomor resi yang muncul berulang (964) menunjukkan pengiriman ulang yang tidak dimodelkan secara khusus. Ketujuh, SMOTENC diimplementasikan sendiri sehingga belum dibandingkan dengan pustaka imbalanced-learn; verifikasi tersebut perlu dilakukan pada penelitian lanjutan.

KESIMPULAN

Random Forest memiliki kemampuan pengurutan risiko tertinggi secara numerik (ROC-AUC 0,745; PR-AUC 0,370, sekitar 4,3 kali nilai acak), tetapi tidak berbeda signifikan dari Decision Tree, dan keduanya jelas lebih baik daripada Logistic Regression. SMOTENC tidak meningkatkan ROC-AUC maupun PR-AUC; ia menggeser titik operasi sehingga lebih banyak kasus terdeteksi dengan lebih banyak peringatan keliru, setara dengan *class_weight*. *Hyperparameter tuning* memberi peningkatan PR-AUC sekitar 0,03. Area tujuan, hari, dan siklus merupakan prediktor utama, sedangkan berat tertagih berperan kecil. Karena precision hanya 0,236 dan kinerja menurun pada validasi temporal (PR-AUC 0,143), model layak dipakai sebagai alat prioritas pengawasan, bukan keputusan otomatis.

Penelitian selanjutnya disarankan (1) memakai data multi-bulan dan beberapa cabang dengan validasi temporal; (2) menambahkan fitur seperti jarak, kurir, dan riwayat masalah per area; (3) menguji XGBoost atau LightGBM serta pendekatan *cost-sensitive* dan penyetelan ambang; (4) menetapkan definisi label bersama pihak operasional; dan (5) mengevaluasi dampak operasional model melalui uji coba pada *dashboard* cabang.

DAFTAR PUSTAKA

- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/jair.953>
- Davis, J., & Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)* (pp. 233–240). ACM. <https://doi.org/10.1145/1143844.1143874>
- de Leeuw, J., Bukhsh, Z., & Zhang, Y. (2023). *Parcel loss prediction in last-mile delivery: Deep and non-deep approaches with insights from Explainable AI* [Preprint]. arXiv. <https://arxiv.org/abs/2310.16602>
- Elor, Y., & Averbuch-Elor, H. (2022). *To SMOTE, or not to SMOTE?* [Preprint]. arXiv. <https://arxiv.org/abs/2201.08528>
- Erlin, E., Desnelita, Y., Nasution, N., Suryati, L., & Zoromi, F. (2022). Dampak SMOTE terhadap kinerja Random Forest classifier berdasarkan data tidak seimbang. *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, 21(3), 677–690. <https://doi.org/10.30812/matrik.v21i3.1726>
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874.

- <https://doi.org/10.1016/j.patrec.2005.10.010>
- Hayati, E., Nurjanah, F., & Kurnia Laili, F. (2024). Unbalanced timeliness of financial reporting data classification using Random Forest with SMOTE. *Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, 9(2), 132–136. <https://doi.org/10.25139/inform.v9i2.8327>
- Lemaître, G., Nogueira, F., & Aridas, C. K. (2017). Imbalanced-learn: A Python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17), 1–5. <https://jmlr.org/papers/v18/16-365.html>
- Lim, S. F. W. T., Wang, Q., & Webster, S. (2023). Do it right the first time: Vehicle routing with home delivery attempt predictors. *Production and Operations Management*, 32(4), 1262–1284. <https://doi.org/10.1111/poms.13926>
- Micci-Barreca, D. (2001). A preprocessing scheme for high-cardinality categorical attributes in classification and prediction problems. *ACM SIGKDD Explorations Newsletter*, 3(1), 27–32. <https://doi.org/10.1145/507533.507538>
- Pambudi, H. K., Kusuma, P. G. A., Yulianti, F., & Julian, K. A. (2020). Prediksi status pengiriman barang menggunakan metode machine learning. *Jurnal Ilmiah Teknologi Informasi Terapan*, 6(2), 100–109. <https://doi.org/10.33197/jitter.vol6.iss2.2020.396>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. <https://jmlr.org/papers/v12/pedregosa11a.html>
- Richardson, E., Trevizani, R., Greenbaum, J. A., Carter, H., Nielsen, M., & Peters, B. (2024). The receiver operating characteristic curve accurately assesses imbalanced datasets. *Patterns*, 5(6), 100994. <https://doi.org/10.1016/j.patter.2024.100994>
- Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLOS ONE*, 10(3), e0118432. <https://doi.org/10.1371/journal.pone.0118432>
- Strobl, C., Boulesteix, A.-L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC Bioinformatics*, 8, Article 25. <https://doi.org/10.1186/1471-2105-8-25>
- van den Goorbergh, R., van Smeden, M., Timmerman, D., & Van Calster, B. (2022). The harm of class imbalance corrections for risk prediction models: Illustration and simulation using logistic regression. *Journal of the American Medical Informatics Association*, 29(9), 1525–1534. <https://doi.org/10.1093/jamia/ocac093>