Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

Predicting Cash-on-Delivery Delivery Failures on an Indonesian Marketplace: A Leakage-Aware, Cost-Sensitive Machine Learning Evaluation

Qurrotul Aini*✉, Mohammad Al-Habsie✉, Muhammad Rafi Habibie✉, Nia Kumaladewi✉
Information Systems Study Program, Faculty of Science and Technology, UIN Syarif Hidayatullah Jakarta, Indonesia

Abstract: Cash on delivery (COD) remains a leading payment option in Indonesian e-commerce, but failed deliveries shift costs to sellers after dispatch. This study examines delivery failure among 11,538 COD orders from one Shopee store between December 2023 and November 2025 and evaluates whether checkout risk scores can support seller decisions. A leakage-aware approach excluded payment and shipping-fee variables that were zero for all cancelled orders, removed 172 orders without final status, and recovered missing months from file names. Logistic regression, random forest, and LightGBM were tuned using December 2023–April 2025 data and evaluated on May–November 2025 data. Class weighting was compared with SMOTENC. Delivery failure reached 4.9% outside Java, compared with 1.1% in Java (odds ratio 4.65; 95% CI 3.59–6.04). On the time-based test set, comprising 3,917 orders and 74 failures, models achieved PR-AUC values of 0.040–0.046, compared with 0.027 for an outside-Java rule and 0.019 prevalence. However, improvements were small, and differences among models were within bootstrap uncertainty. Calibrated Brier scores (0.0185) did not outperform a constant-rate forecast (0.0186). Random cross-validation produced optimistic results, while geographic effects weakened between training and test periods. Cost analysis showed no reliable savings under the base-case assumptions, with potential benefits mainly when contact costs were very low. These findings suggest that machine learning risk scores are more suitable as low-cost screening tools within seller decision dashboards than as automatic intervention rules. The study's conclusions are limited to one store and a 24-month observation period.
Keywords: cash on delivery, delivery failure prediction, class imbalance, cost-sensitive learning, data leakage, e-commerce logistics, decision support

1. INTRODUCTION

Cash on delivery (COD) lets a buyer pay the courier when the parcel arrives. In Indonesia it remains widespread: a survey-based study of 546 users describes COD as still dominating consumer-to-consumer e-commerce payments [1], and studies of buyer motives point to personal, promotional and social factors [2] and to the appeal of paying only when the goods arrive [3]. A descriptive study in Aceh reported that more than 60% of the online purchases it sampled were paid by COD [3]. For sellers on marketplaces such as Shopee, COD therefore matters commercially, and its operational consequences are correspondingly large.

COD also relocates risk. A prepaid order that is never paid is cancelled by the platform before anything ships, whereas a COD order can be refused at the door after the parcel

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

has travelled; the same study of COD practice notes disputes between couriers and buyers who do not understand the procedure [3]. In the order export analysed here, the total cancellation share is almost identical for COD and non-COD orders (13.4% versus 13.8%), but the mechanism differs: all COD delivery failures happen after dispatch, while non-COD cancellations include orders that were never paid. Sellers also work with imperfect data. Marketplace exports are heavily imbalanced (delivery failure affects 2.18% of COD orders) and contain fields that are written only after the outcome is known, which makes naive prediction look better than it is.

Machine learning has been applied to e-commerce order outcomes. Studies on Shopee data use support vector machines with SMOTE oversampling and k-fold validation to predict cancellations [4], compare random forest and XGBoost with SMOTE on a 20,848-transaction data set and attach a cost simulation [5], examine payment and shipping options against cancellations for a merchant [6], or train multi-class models for cancellations and returns [7], [8]. In logistics, failed first delivery attempts have been shown to be predictable from routing and demand attributes and to matter for routing cost [9], and service-failure prediction has been studied in supply-chain networks [10].

Read critically, the closest studies differ from the present problem in ways that matter to a seller. Maulidia and Yustanti [4] and Zhafira and Yunita [5] model cancellation of all causes, so a buyer who cancels before dispatch and a buyer who refuses a parcel at the door fall into one class and the post-dispatch loss specific to COD is not isolated. Their headline results are accuracy figures obtained with oversampling (95.57% under k-fold validation [4]; 86.63% with 37% recall for XGBoost [5]); when the minority class is a few percent of orders, accuracy is dominated by the majority class, and a recall of 37% means that most cancellations are still missed. To our reading, neither describes a time-ordered evaluation, and [4] lists total payment among its influential factors, a field that in our export is zero for every cancelled order and therefore encodes the outcome; whether this applied to their data we cannot determine. Of the studies above, only [5] attaches a cost simulation, the step that links a score to a seller's decision. Logistics studies on failed deliveries [9], [10] predict failure from route and package attributes that a marketplace seller does not hold, and use the prediction for re-routing rather than for order confirmation. To our knowledge, no study combines post-dispatch COD failure, an explicit leakage audit, time-ordered validation with drift diagnostics, calibration and cost-sensitive decisions with uncertainty in one evaluation; this combination is the gap addressed here.

These studies leave five gaps. First, they predict cancellation of all causes, which pools pre-dispatch cancellations with post-dispatch failures; the COD-specific failure that actually costs logistics money is not isolated. Second, validation is typically random k-fold and rarely time-ordered so performance under drift, and the gap between random and time-ordered estimates, is not reported. Third, marketplace exports contain outcome-dependent fields in our data, total payment and several shipping-fee fields are exactly zero for every cancelled order and a study that uses such fields can overstate accuracy [11]; we cannot verify how each cited study handled them, but a leakage audit is rarely reported. Fourth, few studies translate scores into a decision with stated costs and uncertainty; accuracy or recall alone does not show whether an intervention pays. Fifth, probability quality is rarely examined: a score compared with a monetary break-even needs calibrated absolute risks and not only a ranking, yet calibration after oversampling or under drift is seldom reported [14].

We therefore treat COD delivery failure as a separate, rare-event prediction problem with a leakage audit, time-ordered validation and a decision-cost layer. We compare an interpretable logistic regression, a random forest and gradient boosting (LightGBM [12]) against two transparent baselines (an "outside Java" rule and a province-rate lookup), compare class weighting with SMOTENC oversampling [13] applied only inside training

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

folds, and report uncertainty by bootstrap. We then convert calibrated probabilities into flag-or-not decisions using the order's shipping-cost estimate as a loss proxy, with every cost assumption explicit and varied in a sensitivity grid.

The contributions are: (1) evidence that COD shifts cancellation loss to the post-dispatch stage and that delivery failure is concentrated geographically (odds ratio 4.65 outside versus inside Java); (2) a documented leakage-aware, reproducible pipeline for marketplace order exports, with all non-final and missing-date cases accounted for; (3) an honest benchmark of three models and two imbalance strategies on a time-based split, with bootstrap intervals, calibration and robustness checks, showing that model gains over simple baselines are small; (4) a cost-sensitive decision analysis with explicit assumptions and sensitivity ranges; and (5) a design for placing the risk score as a decision layer on an existing Tableau Public dashboard. The novelty is therefore evaluative and methodological rather than algorithmic: we propose no new model, and we report the conditions under which standard models do and do not help a seller.

2. RELATED WORK

COD and its operational risk. Research on COD in Indonesia is largely behavioural. Purwandari et al. studied why users might switch from COD to e-payment, using responses from 546 users, and report that COD still dominates C2C e-commerce payments [1]. Vandiny et al. examined the reasons customers choose COD and found personal factors to be the most dominant [2]. Maisyura et al. describe COD as widely used and point to courier–buyer conflicts when buyers do not understand the procedure [3]. These studies explain demand for COD but do not quantify or predict the delivery failures it produces.

Order cancellation prediction. Maulidia and Yustanti predict transaction cancellations for a Shopee store with a support vector machine, SMOTE and PCA, report 95.57% accuracy under k-fold validation, and name total payment and estimated shipping-fee deduction among influential factors [4]. Zhafira and Yunita compare random forest and XGBoost with SMOTE on a 20,848-transaction data set, report 86.63% accuracy and 37% recall for XGBoost, find shipping option, estimated shipping cost and geographic location to be key factors, and add a cost simulation [5]. Jampur and Christmantara relate payment method and shipping option to cancellations for a Shopee merchant [6]. Sarı et al. and Irmak apply multi-class and feature-weighting approaches to cancellations and returns [7], [8]. Relative to these works, we separate post-dispatch COD failure from other cancellations, use a time-ordered split, audit leakage explicitly and report uncertainty.

Delivery failure in logistics. Lim et al. show that failed delivery attempts can be predicted from package- and route-level attributes using retailer data from South America, report that up to 20% of business-to-consumer deliveries fail at the first attempt, and use the predictions inside routing, including a field experiment [9]. Sharma et al. study data models for service-failure prediction in supply-chain networks [10]. Our setting differs: the seller, not the carrier, is the decision maker, and the intervention is an order-level confirmation rather than re-routing.

Imbalance, evaluation and leakage. Oversampling with SMOTE is a common remedy for rare outcomes [13], but imbalance corrections can distort predicted risks and harm calibration even when discrimination is unchanged [14]. For rare events the precision–recall curve is more informative than the ROC curve [15], so PR-AUC is our primary metric. Leakage—using information that would not be available at prediction time—is a documented cause of over-optimistic results in applied machine learning [11]. We apply these lessons by fitting SMOTENC only inside training folds, calibrating probabilities explicitly, reporting PR-AUC with bootstrap intervals and auditing outcome-dependent fields.

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

3. RESEARCH METHODOLOGY

The research methodology section describes the systematic procedures employed to conduct the study, including data collection, data preprocessing, analytical methods, model development, evaluation, and robustness testing. The overall research workflow is presented in Figure 1.

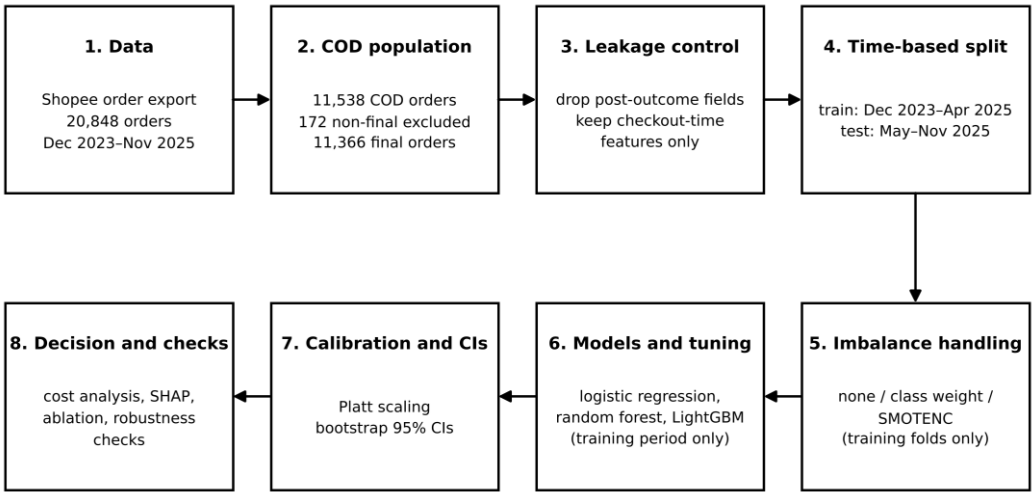

Figure 1. Research workflow, from the marketplace export to cost analysis, interpretation and robustness checks. All tuning, calibration and threshold selection use the training period only.

3.1. Data and study population

Fig. 1 gives an overview of the workflow. The data are the complete order export of one Shopee store for 24 months (December 2023–November 2025): 20,848 orders and 19 columns, semicolon-delimited. Buyer identity is absent; location is limited to city/regency and province. 11,538 orders were paid by COD and 9,310 by other methods. The target is binary: an order is a delivery failure if its cancellation-reason text contains the marketplace's failed-delivery phrase (Indonesian *"Pengiriman gagal"*). This label covers 251 COD orders (2.18% of COD orders) but only 4 of 9,310 non-COD orders (0.04%), so the modelling population is restricted to COD.

Three data issues were handled explicitly. (i) 250 orders (of which 172 COD) had a non-final status (in transit, delivered but still open to return requests); all are dated November 2025, so they are excluded from modelling and reported as right-censored. The analysis population is therefore 11,366 COD orders with a final status (251 failures, 2.21%). (ii) 1,980 orders lack a creation timestamp (all of December 2024 and July 2025). For the orders that have one, the timestamp month equals the month encoded in the source file name in 100.0% of cases, so the file-name month is used as the time index for every order and no month is dropped. (iii) The population is split by time before any modelling: training = December 2023–April 2025 (7,449 orders, 177 failures, 2.38%); test = May–November 2025 (3,917 orders, 74 failures, 1.89%). The split point was fixed in advance to leave about one third of orders and more than 70 events for testing. Table 1 summarises the cohort.

Table 1. Cohort construction and time split (COD orders).

Step	Orders	Delivery failures	Rate (%)
All orders in export	20,848	255	1.22

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

COD orders	11,538	251	2.18
– non-final status (Nov 2025, excluded)	172	–	–
COD, final status (analysis population)	**11,366**	**251**	**2.21**
Training period (Dec 2023–Apr 2025)	7,449	177	2.38
Test period (May–Nov 2025)	3,917	74	1.89

3.2. Leakage control and feature engineering

A leakage audit compared every numeric field across outcomes (Table 2). Total payment, shipping fee paid by the buyer and estimated shipping-fee deduction are exactly zero for all 2,830 cancelled orders, including all 251 COD delivery failures, while total payment is zero for only 0.3% of completed orders. Returned quantity is likewise zero for every failure and is recorded after delivery. These fields (and status and reason text) encode the outcome mechanically and were never used as predictors [11]. Predictors are limited to information that is plausibly known at checkout: province, a Java/outside-Java flag (Banten, DKI Jakarta, West, Central and East Java and DI Yogyakarta versus all other provinces), island group, city versus regency, shipping option parsed into tier and carrier, total weight, quantity, number of product categories, discount amount and flag, estimated shipping cost, shipping cost per kilogram, main product category, month and a peak-season flag (July–October, taken from the descriptive dashboard analysis of the same store). Provinces with fewer than 100 training orders are pooled (13 provinces are kept). The shipping-cost estimate and shipping option are assumed to be fixed at checkout; the evidence for this assumption and its remaining limits are set out after Table 2.

Table 2. Predictors used and fields excluded as outcome-dependent.

Group	Fields	Availability at checkout and basis
Geography (used)	Province (pooled), Java flag, island group, city/regency type	Fixed when the buyer enters the delivery address and places the order.
Logistics (used)	Shipping tier, carrier, total weight, estimated shipping cost, shipping cost per kg	Courier option and estimated fee are selected or shown to the buyer at checkout; weight is derived from listed product weights and quantity. The estimated cost is populated for failed orders (mean Rp33,009), unlike the excluded fee fields. Later revisions cannot be observed in the export.
Order content (used)	Quantity, number of product categories, main product category, discount amount and flag	Cart contents and the discount are fixed when the order is placed.
Temporal (used)	Month of year, peak-season flag (Jul–Oct)	Order month is known at order time. The Jul–Oct window was chosen from whole-period descriptive analysis (mild look-ahead, see text).
Excluded (leakage)	Total payment; shipping fee paid by buyer; estimated shipping-fee deduction; returned quantity; order status; cancellation reason	Zero for every cancelled order or recorded after the outcome; not available at checkout.

Availability at checkout and verification. Availability was assessed for every retained predictor in three steps. First, each field was traced to the point in the order flow at which its value is fixed (last column of Table 2): destination, courier option and estimated shipping fee are determined when the buyer places the order, weight and quantity follow from the cart contents, and the month is the order date itself. Fields that are generated or revised after placement (total payment, buyer-paid and deducted shipping fees, returned quantity, status and reason) were excluded. Second, the export was checked for the

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

signature of an outcome-dependent field, which is zero or missing for cancelled orders, as the three excluded money fields are. The retained shipping-cost estimate does not show it: it is populated for all 251 failed orders (sum Rp8,285,198, mean Rp33,009). Third, we examined how much the conclusions depend on the contested fields. In the ablation of Table 9, removing the logistics group changes PR-AUC by −0.004 for LightGBM, −0.007 for random forest and +0.008 for logistic regression, and removing the temporal group (month and peak-season flag) does not lower PR-AUC for any model, so the findings do not rest on a single contested field.

Two limits remain. The export holds one record per order and no field-level change history, so it cannot show whether the shipping estimate or weight was revised after checkout (for example by courier re-weighing); an estimate revised after the outcome could still carry indirect information. In addition, the peak-season window (July–October) was identified from descriptive analysis of the whole period, which includes test months, so the flag is calendar-based and available at checkout but not strictly out-of-sample. Before deployment, the values shown at checkout should be logged at order creation and compared with the exported values; a mismatch rate close to zero would confirm that the retained fields are free of indirect leakage

3.3. Validation, imbalance handling and tuning

Hyper-parameters were tuned by forward-chaining validation inside the training period only: three validation blocks (August–October 2024, November 2024–January 2025, February–April 2025), each trained on all earlier months, with mean PR-AUC as the criterion. Random search used 16 configurations for logistic regression, 14 for random forest and 18 for LightGBM, separately for each of three imbalance strategies: none, class weighting (balanced) and SMOTENC oversampling (five neighbours, 1:1) fitted inside each training fold after which categorical variables are one-hot encoded. The class-weighted variants were designated primary before test evaluation. Final random forests used 500 trees. Predicted probabilities were calibrated by Platt scaling fitted on stratified five-fold out-of-fold predictions within the training period, which leaves ranking metrics unchanged. No test-period data were used for tuning, calibration or thresholds. Two baselines were defined: the rule "outside Java = risky", and a province-rate lookup (training failure rate per province shrunk towards the overall rate with weight 20).

3.4. Evaluation metrics and inference

The primary metric is PR-AUC (average precision) [15]; we also report ROC-AUC, precision and recall at fixed alert budgets (top 10% and top 20% of orders by score), the Brier score, calibration intercept and slope, and a five-bin reliability curve. Brier scores are compared with two constant forecasts, one at the training failure rate and one at the test prevalence (the lowest Brier score that any constant forecast can reach), and the calibration intercept and slope are the main calibration diagnostics [17]. Uncertainty is quantified with 2,000 bootstrap resamples of test orders (percentile 95% intervals), including paired bootstrap differences between models. For the geographic comparison we use the chi-square test of independence and the odds ratio (OR) with a Woolf interval, Eq. (1), and Wilson intervals for rates.

$$OR = ad / bc; \quad 95\%\ \mathrm{CI} = \exp[\ \ln OR \pm 1.96\ \sqrt{(1/a + 1/b + 1/c + 1/d)}\ ] \qquad (1)$$

where a and b are failed and delivered outside-Java orders and c and d are failed and delivered Java orders. Three supplementary checks were run with hyper-parameters fixed from the training-period tuning: repeated stratified five-fold cross-validation (three repeats) on the whole analysis population as a comparator; a rolling-origin evaluation with three expanding windows (test blocks December 2024–March 2025, April–July 2025, August–November 2025), which partly overlaps months used in tuning and is therefore a robustness check rather than a clean test; and a shipped-only sensitivity analysis that

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

keeps only completed and failed orders. Seed stability was checked with ten random seeds for the stochastic models. Distribution shift between the training and test periods was documented by comparing order volume, the failure rate, the share and failure rate of outside-Java orders, the Java/outside-Java odds ratio (Eq. (1), Woolf interval) and the mean shipping-cost estimate of failed orders (Table 7); its consequences were read from the calibration intercept and slope on the test period.

3.5. Cost-sensitive decision analysis

Let y_i indicate failure of order i and f_i indicate that the seller intervenes (for example a chat confirmation or a prepayment request). If nothing is done, a failure costs L_i, which we proxy by the order's estimated shipping cost. An intervention costs c rupiah per flagged order, prevents a share e of the loss when the order would indeed fail, and costs h rupiah of friction when the order would have been delivered. The realised cost of a policy is Eq. (2); the saving is the cost of doing nothing minus this cost.

$$C(f) = \sum_{i=1}^{n} [\, y_i L_i (1 - e f_i) + f_i (c + (1 - y_i) h) \,] \quad (2)$$

With a calibrated probability p_i, flagging order i reduces expected cost when $p_i \cdot e \cdot L_i - c - (1 - p_i) h > 0$, which is equivalent to Eq. (3). We evaluate five policies on the test period: no intervention, intervene on all COD orders, the outside-Java rule, a global probability threshold chosen to minimise realised cost on training out-of-fold predictions, and the per-order rule of Eq. (3).

$$p_i > (c + h) / (e L_i + h) \quad (3)$$

The parameters e, c and h are **assumptions, not estimates from the data**. The base case (e = 0.5, c = Rp500, h = Rp1,000) was fixed before any cost result was inspected, and a grid of $e \in \{0.3, 0.5, 0.7\}$, $c \in \{100, 250, 500, 1{,}000, 2{,}000\}$ and $h \in \{0, 1{,}000, 3{,}000\}$ (45 scenarios) shows how conclusions depend on them. The data contain no cost price, margin or stock information, so lost margin is not modelled.

3.6. Interpretability, ablation and reproducibility

SHAP values [16] were computed with TreeExplainer for the class-weighted LightGBM model on the test period; values of one-hot columns were summed back to the original variable. Grouped permutation importance (30 repeats) permutes all columns of a feature group jointly. Ablation retrains the models with one feature group removed, with hyper-parameters fixed and five seeds for stochastic models. Analyses were run in Python 3.12.3 with scikit-learn 1.9.1, LightGBM 4.7.0, imbalanced-learn 0.14.2, SHAP 0.54.1, pandas 3.0.6 and SciPy 1.18.1; scripts 01–08 reproduce every table and figure.

4. RESULT AND DISCUSSIONS

4.1. COD shifts loss to the post-dispatch stage and failures are geographically concentrated

Total cancellation is almost the same for COD (13.4%) and non-COD orders (13.8%), but its composition differs (Table 3). All 251 COD failures happen after dispatch, whereas non-COD orders show 447 system cancellations for non-payment (4.8% of non-COD orders), which occur before shipment. Delivery failure is 0.04% for non-COD orders. COD therefore changes where loss occurs rather than whether it occurs. The shipping-cost estimates of the 251 failed orders sum to Rp8,285,198 (mean Rp33,009) over 24 months, about 0.79% of the Rp1,045,550,734 paid by buyers on completed orders: the direct logistics stake is real but modest for a single store.

Table 3. Cancellation composition of COD and non-COD orders (all orders, n = 20,848).

Outcome	COD n	COD %	Non-COD n	Non-COD %
Orders	11,538	100.0	9,310	100.0

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

Cancelled (all causes)	1,544	13.4	1,286	13.8
– delivery failed (after dispatch)	251	2.18	4	0.04
– unpaid, system cancellation (before dispatch)	0	0.00	447	4.80
– buyer-initiated cancellation	1,195	10.36	717	7.70
– seller-side cancellation	97	0.84	115	1.24
– other	1	0.01	3	0.03

Stage of buyer- and seller-initiated cancellations is not recorded in the export; delivery failure is post-dispatch and non-payment is pre-dispatch by definition.

Failure is strongly concentrated outside Java (Table 4). Among all COD orders the failure rate is 4.95% outside Java (159/3,215) versus 1.11% in Java (92/8,323); $\chi^2 = 160.7$ (1 d.f.), $p < 0.001$, OR = 4.65 (95% CI 3.59–6.04). Restricting to final-status orders gives the same picture (OR = 4.77, 95% CI 3.68–6.18). By island group (Fig. 2), failure ranges from 1.12% in Java to 8.36% in Sulawesi, with Bali–Nusa Tenggara at 6.86% and Sumatera at 4.33%; Maluku–Papua has only 24 orders and is not interpretable. Descriptively, pickup-point deliveries fail more often (12.9%, 9/70) and so do orders carried by J&T (9.9%, 20/203, versus 2.1% for SPX); these groups are small and confounded with destination.

Table 4. Delivery-failure rate in Java versus outside Java among COD orders.

Population	Java: fail/orders (rate %, 95% CI)	Outside Java: fail/orders (rate %, 95% CI)	χ^2 (p)	OR (95% CI)
All COD (n = 11,538)	92/8,323 (1.11; 0.90–1.35)	159/3,215 (4.95; 4.25–5.75)	160.7 (<0.001)	4.65 (3.59–6.04)
Final status (n = 11,366)	92/8,250 (1.12; 0.91–1.37)	159/3,116 (5.10; 4.38–5.93)	166.5 (<0.001)	4.77 (3.68–6.18)

Chi-square without continuity correction; with Yates correction the statistics differ by less than 2 units and conclusions are unchanged.

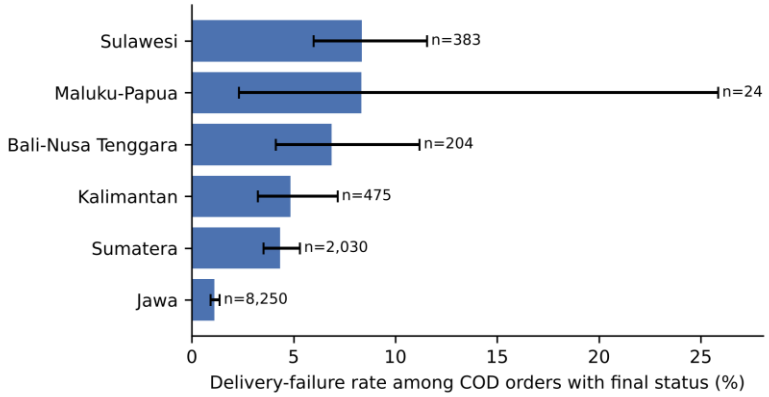

Figure 2. Delivery-failure rate among COD orders with final status by island group (bars: Wilson 95% intervals; n = orders).

4.2. Model comparison on the time-based test set

Table 5 shows test-period performance (prevalence 1.9%). The three class-weighted models reach PR-AUC 0.040 (logistic regression), 0.046 (random forest) and 0.043 (LightGBM), about two to two-and-a-half times the prevalence, and ROC-AUC 0.632–0.658. The outside-Java rule reaches PR-AUC 0.027 and ROC-AUC 0.617; it flags 1,168 of 3,917 orders (29.8%) and catches 39 of 74 failures (recall 0.53, precision 0.033). Models therefore discriminate modestly. Flagging the top 10% of orders by score yields precision

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

of 0.038–0.043 and recall of 0.20–0.23, a lift of about 2.0–2.3. Bootstrap intervals are wide (Fig. 3): for example LightGBM PR-AUC has a 95% interval of 0.027–0.080.

Table 5. Test-period performance (May–November 2025; 3,917 orders, 74 failures; class-weighted models, Platt-calibrated Brier).

Model	PR-AUC (95% CI)	ROC-AUC (95% CI)	Precision @top 10%	Recall @top 10%	Brier
Rule: outside Java	0.027 (0.019–0.035)	0.617 (0.557–0.674)	–	–	–
Province-rate lookup	0.036 (0.023–0.066)	0.624 (0.553–0.695)	0.036	0.19	–
Logistic regression	0.040 (0.025–0.072)	0.632 (0.566–0.699)	0.038	0.20	0.0185
Random forest	0.046 (0.028–0.089)	0.651 (0.584–0.718)	0.043	0.23	0.0185
LightGBM	0.043 (0.027–0.080)	0.658 (0.593–0.720)	0.041	0.22	0.0185

Constant-rate benchmarks: PR-AUC = prevalence = 0.0189; Brier of a constant forecast at the training failure rate = 0.0186. Rule top-10% metrics omitted because a binary score has ties.

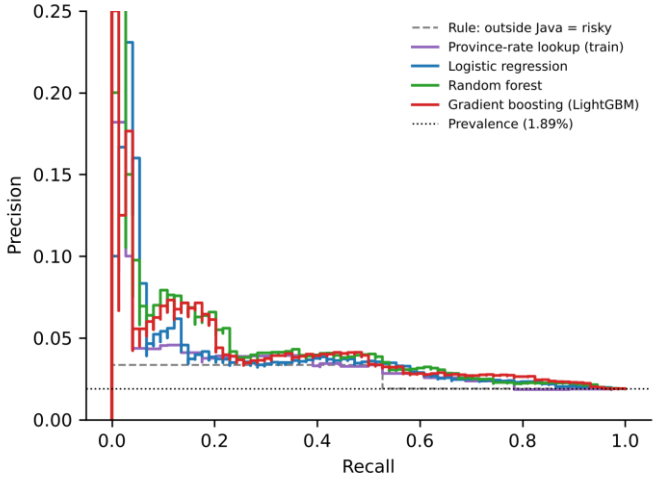

Figure 3. Precision–recall curves on the test period; the dotted line is the failure prevalence.

Paired bootstrap differences show where the evidence lies. Against the outside-Java rule, PR-AUC is higher by 0.013 for logistic regression (95% CI 0.002–0.043), 0.020 for random forest (0.006–0.060) and 0.016 for LightGBM (0.005–0.051); the ROC-AUC gain is 0.042 for LightGBM (0.007–0.078) and 0.034 for random forest (0.000–0.069), but 0.015 for logistic regression (-0.038–0.066). Models are not distinguishable from each other (LightGBM minus logistic regression: 0.003, -0.012–0.022; random forest minus logistic regression: 0.007, -0.011–0.031) nor from the province-rate lookup (LightGBM minus lookup: 0.007, -0.006–0.028). The honest summary is that machine learning improves modestly on the simplest geographic rule, but the extra complexity of ensembles over a regularised logistic regression is not supported by this test set.

Class weighting versus SMOTENC (Table 6) shows no consistent winner. Random forest is best with class weights (PR-AUC 0.046 versus 0.035 with SMOTENC and 0.039 unweighted), logistic regression is highest with SMOTENC on PR-AUC (0.042) but lowest on ROC-AUC (0.614), and LightGBM is best with class weights on PR-AUC. All differences are small relative to the bootstrap intervals above. Both corrections inflate uncalibrated Brier scores (0.085–0.141 for random forest) compared with unweighted models (0.0186),

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

as expected when the class prior is artificially changed [14]; this is why calibration is applied before any decision use.

Table 6. Imbalance strategy comparison on the test period (uncalibrated probabilities; hyper-parameters tuned separately per cell).

Model	Strategy	PR-AUC	ROC-AUC	Brier (raw)
Logistic regression	None	0.036	0.646	0.0187
Logistic regression	Class weight	0.040	0.632	0.1376
Logistic regression	SMOTENC	0.042	0.614	0.1234
Random forest	None	0.039	0.628	0.0186
Random forest	Class weight	0.046	0.651	0.1408
Random forest	SMOTENC	0.035	0.628	0.0851
LightGBM	None	0.036	0.660	0.0188
LightGBM	Class weight	0.043	0.658	0.1281
LightGBM	SMOTENC	0.041	0.640	0.1362

4.3. Calibration and robustness

After Platt scaling the Brier score is 0.0185, 0.0185 and 0.0185 for logistic regression, random forest and LightGBM (95% intervals about 0.0147–0.0225), which equals the 0.0186 of a constant forecast at the training failure rate: the models rank orders modestly but do not improve probability accuracy over that benchmark. Calibration slopes on the test period are 0.55–0.62 and intercepts -1.50 to -1.71 (Fig. 4): predicted risks are too extreme and on average too high. Part of this reflects drift between the training and test periods, which is quantified in Table 7 and discussed below.

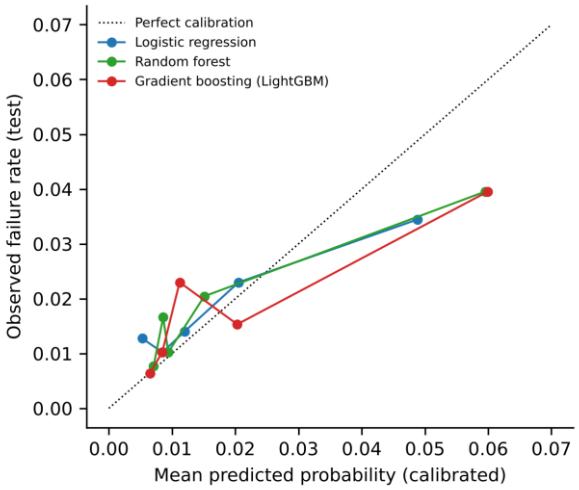

Figure 4. Reliability curves of Platt-calibrated predictions on the test period (five quantile bins).

Drift between training and test periods. Table 7 compares the two periods. Orders per month rose from about 438 to 560 while failures per month stayed near 10, so the failure rate fell from 2.38% to 1.89%. The change was not uniform: among outside-Java orders the failure rate fell from 6.16% (120/1,948) to 3.34% (39/1,168), whereas in Java it rose slightly from 1.04% (57/5,501) to 1.27% (35/2,749). The Java/outside-Java odds ratio therefore fell from 6.27 (95% CI 4.55–8.63) to 2.68 (1.69–4.25), with non-overlapping intervals. Outside-Java orders accounted for 67.8% of failures in training but 52.7% in the test period although their share of orders rose from 26.2% to 29.8%, and the mean shipping-cost estimate of failed orders fell from Rp38,339 to Rp20,259. The geographic

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

signal on which the rule and, to a large extent, the models rely (Section 4.4) thus weakened out of time. This combines a shift in the predictor distribution with a change in the relation between predictors and outcome (concept drift [18]), and it is the most likely reason why every method scores lower on the test set than on random folds and why the calibration intercept is negative and the slope below one.

Table 7. Training versus test period: order volume, failure rates and geographic effect (COD orders with final status).

Quantity	Training (Dec 2023–Apr 2025)	Test (May–Nov 2025)
COD orders with final status	7,449	3,917
Orders per month (17 / 7 months)	438	560
Delivery failures (per month)	177 (10.4)	74 (10.6)
Failure rate (%)	2.38	1.89
Share of orders outside Java (%)	26.2 (1,948)	29.8 (1,168)
Failure rate outside Java, % (fail/orders)	6.16 (120/1,948)	3.34 (39/1,168)
Failure rate in Java, % (fail/orders)	1.04 (57/5,501)	1.27 (35/2,749)
Odds ratio outside vs. in Java (95% CI)	6.27 (4.55–8.63)	2.68 (1.69–4.25)
Share of failures outside Java (%)	67.8	52.7
Mean shipping-cost estimate of failed orders (Rp)	38,339	20,259

Odds ratios from Eq. (1) with Woolf intervals; the interval for the test period lies entirely below that of the training period. Monthly figures divide by 17 and 7 months.

Why the probabilities add nothing over the base rate. Three factors explain the poor calibration quality. First, the signal is weak relative to the noise floor of the Brier score. A constant forecast at the test prevalence scores $\pi(1 - \pi) = 0.0185$, and a model can beat it only through the resolution term of the Brier decomposition. As an illustration, take the LightGBM top-10% queue, which holds about 16 failures among 392 orders (4.1%) against 1.6% among the remaining 3,525 orders; even if both groups were forecast perfectly, the resolution would be about 0.00005, a Brier skill score of at most 0.3%, whereas the bootstrap interval of the Brier score spans 0.0147–0.0225. No recalibration method can create resolution that the ranking does not contain, so the tie with the constant forecast follows from the modest PR-AUC and not from a faulty calibrator. Second, Platt scaling was fitted on out-of-fold predictions within the training period, so it corrects only miscalibration that is already present there; it cannot absorb the shift between periods, which is why the test slope (0.55–0.62) and intercept (−1.50 to −1.71) show risks that are too extreme and too high [17]. Third, with 177 training and 74 test failures the calibration parameters are themselves imprecise, and five-bin reliability curves are noisy.

Implications for decision-making. The probabilities should therefore not be read as absolute risks. Eq. (3) compares each probability with break-even levels of 4.8–25%, whereas the calibrated test probabilities have a 95th percentile of only 5.5–7.5%; because the slope is below one, a slope-corrected probability would be pulled further towards the base rate of 1.9% and fewer orders would clear the break-even level, so the per-order rule would converge to "do nothing", which is what the near-zero savings in Table 10 show. We recommend (a) using a rank-based alert budget, for example the top 10% of orders, whose precision and lift are observed directly, rather than a probability cut-off; (b) re-estimating the calibration intercept and slope on the most recent months before any probability is shown to a seller; and (c) reporting the Brier skill score against a constant forecast next to PR-AUC, so that a score whose probabilities carry no information is not used for absolute-risk decisions.

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

Table 8. Robustness of PR-AUC (ROC-AUC in parentheses) across evaluation designs.

Design (prevalence)	Logistic regression	Random forest	LightGBM	Outside-Java rule
Time-based test (1.89%)	0.040 (0.632)	0.046 (0.651)	0.043 (0.658)	0.027 (0.617)
Stratified 5-fold CV ×3 (2.21%)	0.062 (0.728)	0.061 (0.725)	0.063 (0.728)	0.040 (0.684)
Rolling origin, pooled (1.90%)	0.048 (0.683)	0.047 (0.697)	0.044 (0.694)	0.030 (0.648)
Shipped-only test (2.09%)	0.048 (0.640)	0.055 (0.657)	0.045 (0.661)	0.030 (0.621)

CV and rolling-origin use hyper-parameters tuned on the training period; rolling-origin blocks partly overlap months used in tuning, so it is a robustness check rather than a clean test.

Table 8 shows that conclusions are stable in direction but not in level. Random stratified cross-validation yields PR-AUC of 0.062–0.063 for the models, higher than the 0.040–0.046 of the time-based test and with the outside-Java rule at 0.040; random folds mix past and future months, so a model can be trained on orders that occur after those it is scored on, a form of temporal leakage inherent in random validation that overstates performance on later orders [19]. The gap is not caused by leakage in the time-based test: tuning, calibration and thresholds used only the training period, and the test period was used once for comparison. Part of the gap is mechanical, because PR-AUC scales with prevalence (2.21% in the pooled data against 1.89% in the test period); relative to prevalence the models reach about 2.8 times in random cross-validation, 2.3–2.5 times in rolling origin and 2.1–2.4 times in the time-based test, and the outside-Java rule 1.8 against 1.4 times. The remainder reflects the shift documented in Table 7. The rolling-origin evaluation pools 6,356 orders and 121 failures and again ranks all three models above the rule (0.048, 0.047 and 0.044 versus 0.030), although block-wise PR-AUC varies from 0.026 to 0.092. Restricting to shipped orders (test: 3,545 orders, 74 failures) gives the same ordering. Over ten seeds, LightGBM PR-AUC on the test period ranges from 0.039 to 0.063 (SD 0.007) and random forest from 0.044 to 0.050 (SD 0.002); differences among models smaller than this are not interpretable.

Periodic evaluation for evolving order data. Because the geographic effect changed within a year, a model fitted once cannot be assumed to stay valid. We propose the following routine for a seller who scores orders continuously; it is a proposal and was not evaluated here. (i) Each month, once orders have reached a final status, recompute the quantities of Table 7 on the newly resolved COD orders, together with the population stability index of the main predictors. (ii) On a trailing window, compute PR-AUC with a bootstrap interval, the calibration intercept and slope, and the Brier skill score against a constant forecast. With about 10 failures per month in this store, single-month estimates are too noisy, so a window of about six months (roughly 60 failures) is a practical minimum. (iii) Re-estimate the calibration intercept and slope on that window every month, which is cheap and addresses the most visible symptom. (iv) Retrain, comparing an expanding with a sliding training window, when the slope leaves a tolerance band (for example 0.8–1.2) or when the PR-AUC interval no longer clears that of the regional rule in two consecutive evaluations. The rolling-origin design of Table 8 is the offline analogue of this routine, and the thresholds above are illustrative and should be set from a store's own data.

4.4. Interpretation and ablation

SHAP importance for the class-weighted LightGBM model (Fig. 5a) is led by the shipping-cost estimate (23.7% of total mean absolute SHAP), the island group (17.9%), the Java flag (12.5%), shipping cost per kilogram (12.3%), month (9.4%) and total weight (6.2%). Grouped, logistics variables carry 44.4% of importance, geography 34.5%, temporal

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

variables 13.1% and order content 8.0%; discount variables are essentially unused because only 31 COD orders carry a discount. Orders outside Java receive a mean SHAP contribution of 0.28 for the Java flag, against -0.11 inside Java, and the SHAP contribution of the shipping-cost estimate rises with the estimate (correlation 0.68; Fig. 5b). The shipping-cost estimate largely encodes distance and destination, so geography and logistics are partly the same signal, consistent with the finding of [5] that shipping option, shipping-cost estimate and geographic location are key factors for cancellations. Importance describes the model, not causes.

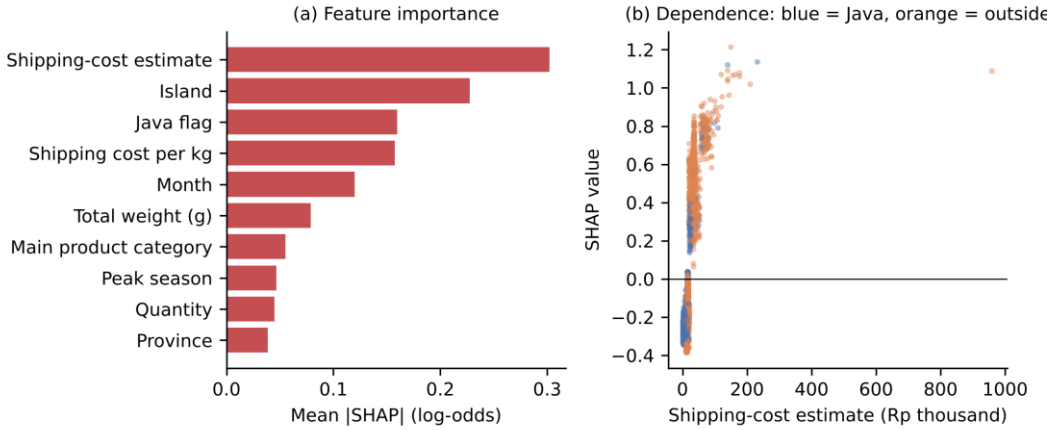

Figure 5. SHAP analysis of the class-weighted LightGBM model on the test period: (a) mean absolute SHAP by grouped variable; (b) SHAP value of the shipping-cost estimate against its value.

Grouped permutation importance agrees (decrease in test PR-AUC, mean ± SD over 30 permutations): geography 0.011 ± 0.004, logistics 0.011 ± 0.007, order content 0.006 ± 0.004 and temporal -0.005 ± 0.007; in ROC-AUC terms geography costs 0.058 and logistics 0.042. Ablation (Table 9) shows that removing any single group changes PR-AUC by about 0.01 or less for LightGBM and random forest, comparable to seed variation, so no single group is indispensable because the groups are redundant. A geography-only model reaches 0.032 (LightGBM) and the Java flag alone 0.027, both below the full models (0.046), which suggests that the combination of location and logistics adds some ranking information.

Table 9. Feature-group ablation on the test period: PR-AUC (± SD over five seeds; ROC-AUC in parentheses).

Feature set	LightGBM	Random forest	Logistic regression
All features	0.046 ± 0.005 (0.660)	0.046 ± 0.001 (0.652)	0.040 (0.632)
− Geography	0.042 ± 0.002 (0.664)	0.047 ± 0.004 (0.658)	0.036 (0.646)
− Logistics	0.042 ± 0.003 (0.647)	0.039 ± 0.001 (0.647)	0.048 (0.613)
− Order content	0.037 ± 0.001 (0.651)	0.041 ± 0.002 (0.652)	0.032 (0.644)
− Temporal	0.048 ± 0.005 (0.645)	0.047 ± 0.001 (0.651)	0.041 (0.633)
Geography only	0.032 ± 0.000 (0.620)	0.033 ± 0.001 (0.624)	n/a
Java flag only	0.027 ± 0.000 (0.617)	0.027 ± 0.000 (0.617)	n/a

Hyper-parameters fixed from tuning with all features. n/a: model not estimable with no numeric inputs.

Operational reading of the explanations. For a seller, the SHAP and ablation results mean three things. First, the main drivers are order attributes that the seller cannot change at checkout (destination, courier, weight), so the explanation does not point to a fix of the order itself; its use is to decide which orders deserve a confirmation message before

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

dispatch. Second, the shipping-cost estimate is both the strongest driver of predicted risk (23.7% of SHAP importance) and the loss proxy in Eq. (3): costly, remote orders have a higher predicted risk and a lower break-even probability (4.8% at Rp60,000 against 25.0% at Rp10,000), so the orders that qualify for confirmation are those where risk and stake are both high, which keeps the queue short (the per-order rules flagged 126–215 orders over seven months, about 18–31 per month). Third, because geography and logistics carry largely the same signal and no group is indispensable, a seller can obtain most of the ranking benefit from the transparent regional rule or a regularised logistic regression and does not need to maintain an ensemble in order to act on these explanations. Importance describes the model and not causes: the higher failure rates of pickup-point deliveries and of J&T (Section 4.1) should not be read as courier effects, because they are confounded with destination.

4.5. Cost-sensitive decision analysis

In the test period the 74 failed orders carry a shipping-cost loss of Rp1,499,200 (mean Rp20,259). Under the base assumptions, even a perfect predictor that flags exactly the failures would save Rp712,600 (47.5% of that loss, about Rp181,925 per 1,000 orders), because each intervention is only half effective. Table 9 shows realised savings of the policies. Intervening on all COD orders loses Rp5,051,900, and the outside-Java rule loses Rp1,119,850 (95% interval −1,325,858 to −877,290) because it flags 1,168 orders, most of which would have been delivered. Global-threshold policies chosen on training predictions also lose money on the test period (LightGBM −358,850), whereas the per-order rule of Eq. (3), which only flags orders whose calibrated risk exceeds a break-even level, is close to zero (logistic regression +8,650; LightGBM −85,350; random forest −86,100) with intervals that include zero. No policy shows a reliable positive saving in the base case.

Table 10. Base-case policy results on the test period (e = 0.5, c = Rp500, h = Rp1,000; savings in rupiah versus no intervention).

Policy	Flagged	Failures caught	Saving (Rp)	95% bootstrap interval
Intervene on all COD orders	3,917	74	−5,051,900	−5,286,504 to −4,812,274
Rule: outside Java	1,168	39	−1,119,850	−1,325,858 to −877,290
Logistic regression, threshold	317	11	−213,600	−363,250 to −40,179
Random forest, threshold	321	17	−121,850	−291,201 to +65,177
LightGBM, threshold	500	18	−358,850	−550,281 to −154,220
Logistic regression, per-order	126	6	+8,650	−138,525 to +179,351
Random forest, per-order	215	8	−86,100	−239,762 to +95,749
LightGBM, per-order	202	7	−85,350	−232,360 to +93,304

No-intervention loss = Rp1,499,200; oracle (flag exactly the failures) saving = Rp712,600. Thresholds chosen on training out-of-fold predictions.

The reason is the break-even risk implied by Eq. (3). At the base costs an order must have a failure probability above 25.0% if its shipping-cost estimate is Rp10,000, 15.1% at Rp17,885 (the mean for delivered COD orders), 8.6% at Rp33,009 (the mean for failures over the full period) and 4.8% at Rp60,000. The calibrated test probabilities have a 95th percentile of only 5.5–7.5% and a maximum of 10.2–17.2%, so few orders qualify. Table

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

11 and Fig. 6 show how this changes with the assumptions. When contact is almost free and causes no friction (c = Rp100–250, h = 0), flagging pays off, but then the simple outside-Java rule captures most of the saving and the models add only a modest margin that was not tested for significance. Across the 45 scenarios, the LightGBM per-order rule saves money in 10, the logistic-regression per-order rule in 21, the outside-Java rule in 8 and flagging everything in 4. Taking the best of the six model-based policies (three models, two decision rules) in each scenario gives savings in 31 scenarios, but that figure is optimistic because it selects the maximum after the fact.

Table 11. Selected sensitivity scenarios (e = 0.5): net saving in Rp thousand on the test period.

c / h (Rp)	Outside-Java rule	Flag all COD	Logistic reg., per-order	LightGBM, per-order	LightGBM flagged (n)
100 / 0	+476	+358	+481	+509	1333
250 / 0	+301	−230	+320	+378	818
500 / 0	+9	−1,209	+150	+150	596
500 / 1,000	−1,120	−5,052	+9	−85	202
100 / 1,000	−653	−3,485	−3	−43	349
1,000 / 1,000	−1,704	−7,010	+5	−67	124
500 / 3,000	−3,378	−12,738	−44	−57	59

Full 45-scenario grid in the supplementary results (T_cost_grid.csv).

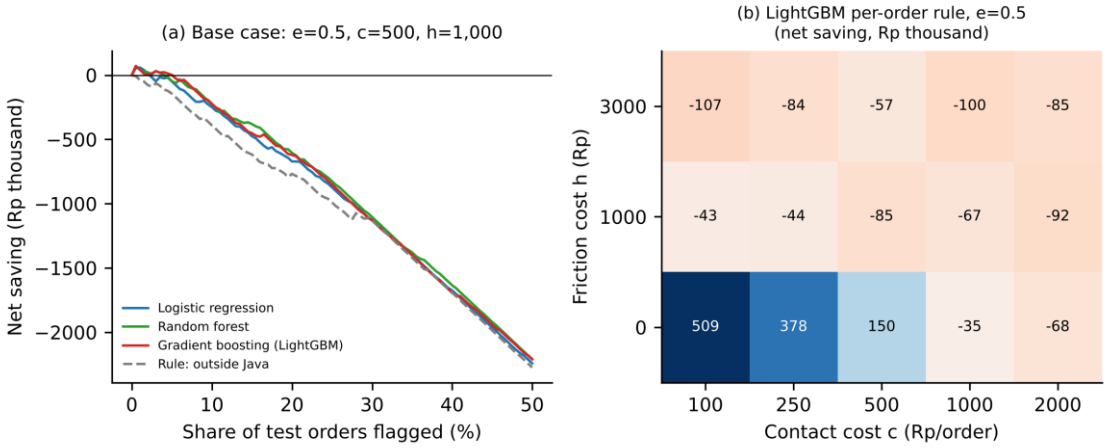

Figure 6. Cost analysis: (a) net saving when flagging the top share of test orders by each score (base case); (b) LightGBM per-order rule across contact cost c and friction cost h (e = 0.5).

Feasibility for sellers. Expressed per month, the test period contained about 560 COD orders and 10.6 failures. A confirmation queue of the top 10% of orders by score would hold about 56 orders per month and, at the observed recall of 0.20–0.23, about 2 of the 10.6 monthly failures. With the test-period mean loss of Rp20,259 and e = 0.5, the expected avoided loss is roughly Rp23,000 per month for LightGBM (a back-of-envelope figure that treats caught failures as average-sized), against about Rp28,000 per month of contact cost at c = Rp500 before any friction cost h, so the queue does not pay for itself at the assumed costs, in line with Table 10. Scoring is technically feasible (weekly offline scoring, Section 4.6), but the economic case depends on a contact cost close to zero (an automated message), on losses larger than the shipping-cost proxy (return handling, lost margin, which the export cannot show), or on higher-risk subsets. A seller should therefore start with automated, low-friction confirmation of the highest-scoring orders and measure its effect before changing any process; a blanket restriction of COD outside Java would

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : https://doi.org/10.58602/itsecs.v5i1.505
Page 210-228

burden buyers in regions where COD may be the main payment option and, in these data, would cost more than it saves.

4.6. A decision layer for the Tableau Public dashboard

The seller already maintains a Tableau Public dashboard of the store's orders (https://public.tableau.com/app/profile/muhammad.rafi.habibie/viz/shared/6K9NPMGGH) . We propose to add the risk score as a decision layer (Fig. 7); this is a design, not a deployed or validated system. Because Tableau Public does not, to our knowledge, call external model services, scoring is run offline: each week the new COD orders are exported, features are built exactly as in training, calibrated probabilities and expected loss $p_i \cdot e \cdot L_i$ are computed, and a scored table is written and joined to the dashboard data source. Three views follow from the results. A confirmation queue lists orders above the break-even rule of Eq. (3) so that the seller contacts the buyer or requests prepayment before dispatch. A geography view shows failure rates and expected loss by island and province, where the evidence is strongest. A monitoring view compares predicted and observed failure rates and the calibration slope by month; given the drift observed in this study, retraining should be triggered when calibration deteriorates. Since Tableau Public makes published data publicly visible, only aggregated risk bands should be published there, and the order-level queue should stay in a private workbook.

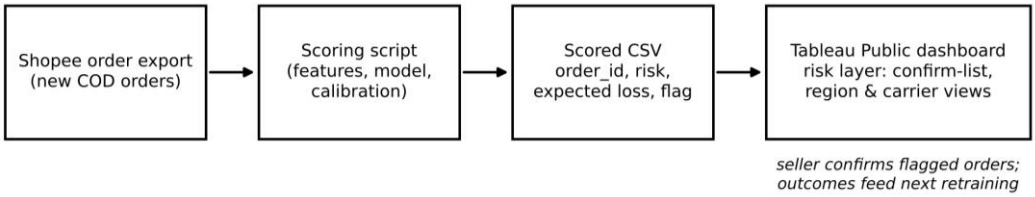

Figure 7. Proposed decision layer: offline scoring feeds a scored table that drives the confirmation queue, geography and monitoring views of the dashboard.

4.7. Discussion

Managerial implications. The most robust finding is geographic: COD orders outside Java fail four to five times as often as those inside Java in odds terms, and this holds in both populations and in the descriptive island breakdown. A transparent regional rule is easy to explain and already captures most of the ranking signal. Because the calibrated probabilities do not beat a constant forecast (Section 4.3), the score should be used to rank orders into an alert budget and not as an absolute risk shown to the seller. The models add a modest, uncertain gain, so the decision to act should rest on cost assumptions rather than on model accuracy: at the contact and friction costs considered here only very cheap interventions, such as an automated message, have a chance of paying for themselves, and blanket confirmation of every outside-Java order does not. A controlled test of the intervention itself (does confirmation actually prevent a failure, and at what cost in lost orders?) is needed to replace the assumed e and h.

Comparison with other research. Prior Shopee studies report accuracy of 95.57% [4] and 86.63% [5] for cancellation prediction. Such figures are not comparable with ours: their target pools all cancellation causes, and accuracy depends heavily on class balance—if the cancelled share in their data is similar to the 13.6% of cancelled orders in our export, always predicting "not cancelled" would already reach about 86% accuracy. Our results are consistent with theirs on which variables matter—shipping option, shipping-cost estimate and location [5] but our performance estimates are modest, which is expected for a rarer outcome evaluated on a time-ordered split with outcome-dependent fields excluded. One study names total payment among its influential factors [4]; in our export

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

that field is exactly zero for every cancelled order, which is why such fields need an explicit audit, although we cannot determine whether that study was affected. In logistics, delivery-attempt prediction has used package- and route-level attributes that a seller does not have [9]; a seller-side confirmation decision has far fewer features to work with.

5. THREATS TO VALIDITY AND LIMITATIONS

External validity. The data are one Shopee store over 24 months, and the product mix is household goods; results may not transfer to other stores, categories or marketplaces. The odds ratio, the size of the drift and the break-even results are properties of this store and period, and no multi-store or multi-marketplace validation was possible with the data available. *Statistical power.* The test period contains only 74 failures, so intervals are wide and model rankings are fragile; ten-seed results for LightGBM span 0.039–0.063. *Drift.* Failure rates, destinations and shipping costs changed between training and test periods, and random cross-validation overstates performance. The odds ratio between regions fell from 6.3 to 2.7 (Table 7); drift is documented for geography, order volume and shipping cost only, and a predictor-by-predictor stability analysis remains to be done. *Missing and censored data.* 1,980 orders lack timestamps (December 2024 and July 2025), so only month-level time information is used and no time-of-day or weekday features are possible; all 172 non-final COD orders are from November 2025, and failures among them would raise the observed rate, so the most recent month is slightly biased. *Label definition.* The label depends on the marketplace's cancellation-reason text; reasons that are mislabelled or changed over time are not detectable. *Leakage and availability.* We assume that the shipping-cost estimate, shipping option and weight are fixed at checkout; if the marketplace revises them afterwards, performance would be overstated. The export has no field-level change history, so this could not be tested directly (see the discussion after Table 2). *Missing business variables.* The export has no buyer identifier, cost price or stock, so we make no claim about repeat customers or margins; the loss proxy covers only the shipping-cost estimate and excludes return handling, damaged goods, lost margin and platform penalties, and the intervention parameters are assumptions. *Analytic choices.* The peak-season flag was defined from descriptive analysis of the whole period, province pooling used the training period only, and several robustness analyses reuse months seen during tuning; the final test period was used once for model comparison. *Multiplicity.* Many comparisons are reported; interval estimates, not p-values, should guide interpretation. *Calibration.* Calibrated probabilities do not improve on a constant forecast and are shifted by drift, so they should not be read as absolute risks.

6. CONCLUSION AND FUTURE WORK

We analysed 11,366 final-status COD orders from one Shopee store to predict delivery failure with a leakage-aware, time-ordered design. COD delivery failure is a post-dispatch loss that is concentrated outside Java (OR 4.65, 95% CI 3.59–6.04). On the held-out period, logistic regression, random forest and LightGBM achieved PR-AUC of 0.040–0.046, modestly above an outside-Java rule (0.027) and about two to two-and-a-half times the failure prevalence, but they were not distinguishable from one another and did not improve probability accuracy over a constant forecast. Random cross-validation overstated performance, the geographic effect weakened between the training and test periods (odds ratio 6.3 to 2.7), and under assumed intervention costs model-guided confirmation was rarely profitable unless contact cost was very low. We recommend using the score as a low-cost screening and monitoring layer on the dashboard, with the break-even rule of Eq. (3) as the decision criterion and a regional rule as a transparent fallback. These findings come from one Shopee store and one 24-month observation period; the odds ratio, the

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

size of the drift and the break-even results are specific to that store, and they should not be generalised to other sellers, product categories or marketplaces before replication.

Future work should (i) validate the pipeline across multiple stores, product categories and marketplaces and over longer periods, applying the monitoring routine of Section 4.3 to each, (ii) add information that is currently missing—buyer history, address quality, order time and courier performance—and test whether it lifts discrimination, (iii) run controlled experiments, in which comparable orders are randomly assigned to confirmation, a prepayment request or no contact, to measure the effectiveness and friction of these interventions so that *e*, *c* and *h* are estimated rather than assumed, (iv) model time to failure and handle the right-censored recent orders, and (v) implement and monitor the scoring layer on the dashboard.

ETHICS AND DATA AVAILABILITY

The data are order records of a single Shopee store and contain no buyer names, contact details or street addresses; location is limited to city/regency and province.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Qurrotul Aini: Investigation (literature review), Validation, Supervision, Funding acquisition, Writing – review & editing. **Mohammad Al-Habsie:** Conceptualization, Methodology, Software, Investigation (data acquisition and leakage audit), Formal analysis, Data curation, Visualization, Writing – original draft. **Muhammad Rafi Habibie:** Investigation (literature review), Data curation (data cleaning and cohort construction), Validation, Visualization, Writing – review & editing. **Nia Kumaladewi:** Methodology (guidance on the visualisation and how the analysis works), Validation, Supervision, Funding acquisition, Writing – review & editing.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

FUNDING STATEMENTS

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. The costs of the research were paid personally by the supervising lecturer, N. Kumaladewi, who is a co-author of this paper.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this work, the authors used generative AI tools to assist with the writing of the manuscript, namely improving language, drafting and revising text, and checking the internal consistency of numbers and citations. Generative AI was not used for the research design, data collection, or the interpretation of results, which were carried out by the authors. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

REFERENCES

[1] B. Purwandari, S. A. Suriazdin, A. N. Hidayanto, S. Setiawan, K. Phusavat, and M. Maulida, "Factors affecting switching intention from cash on delivery to e-payment services in C2C e-commerce transactions: COVID-19, transaction, and technology perspectives," *Emerg. Sci. J.*, vol. 6, Special Issue, pp. 136–150, 2022, doi: 10.28991/esj-2022-SPER-010.

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 5, Number 1, January 2027
DOI : **https://doi.org/10.58602/itsecs.v5i1.505**
Page 210-228

[2] V. S. Vandiny, R. Listiawati, T. Rimenda, and J. Marbun, "The reasons of why customers choose COD as their payment method when shopping online," in *Proc. Int. Conf. Appl. Sci. Technol. Soc. Sci. (iCAST-SS 2022)*, Adv. Soc. Sci., Educ. Humanit. Res., vol. 719, 2022, pp. 535–540, doi: 10.2991/978-2-494069-83-1_95.
[3] Maisyura, C. Sukmawati, R. Dewi, and Arinanda, "Analysis of cash on delivery (COD) payment methods in online shopping transactions in Indonesia," in *Proc. 2nd Int. Conf. Soc. Sci., Political Sci., Humanit. (ICoSPOLHUM 2021)*, Adv. Soc. Sci., Educ. Humanit. Res., vol. 648, 2022, pp. 269–274, doi: 10.2991/assehr.k.220302.040.
[4] R. Maulidia and W. Yustanti, "Implementation of the support vector machine (SVM) algorithm in predicting transaction cancellations at Shopee e-commerce," *J. Emerg. Inf. Syst. Bus. Intell. (JEISBI)*, vol. 6, no. 1, pp. 49–62, 2025, doi: 10.26740/jeisbi.v6i1.64414.
[5] A. Zhafira and F. Yunita, "Analisis prediksi pembatalan pesanan e-commerce menggunakan algoritma Random Forest dan XGBoost untuk mitigasi kerugian finansial," *Indones. J. Multidiscip. Soc. Technol.*, vol. 4, no. 3, pp. 3143–3152, 2026, doi: 10.69693/ijmst.v4i3.12964.
[6] F. X. A. P. Jampur and V. Christmantara, "Analisis dampak metode pembayaran dan opsi pengiriman terhadap cancel order pada merchant toko mainan edukasi anak di Shopee," *J. Sist. Inf.*, vol. 17, no. 1, pp. 70–81, 2025, doi: 10.18495/jsi.v17i1.233.
[7] Z. S. Sarı, H. H. Yurdagül, B. Taşkapı, G. Dağdaş, and M. F. Akay, "Multi-class machine learning models for predicting order cancellations and returns in e-commerce," in *Proc. 9th Int. Symp. Innov. Approaches Smart Technol. (ISAS)*, 2025, doi: 10.1109/ISAS66241.2025.11101817.
[8] T. Irmak, "Analysis of order cancellation rates and feature weighting with machine learning: An e-commerce case study," in *Intelligent and Fuzzy Systems (INFUS 2025)*, vol. 1531, 2025, pp. 763–770, doi: 10.1007/978-3-031-98304-7_82.
[9] S. F. W. T. Lim, Q. Wang, and S. Webster, "Do it right the first time: Vehicle routing with home delivery attempt predictors," *Prod. Oper. Manag.*, vol. 32, no. 4, pp. 1262–1284, 2023, doi: 10.1111/poms.13926.
[10] M. Sharma, T. Glatard, É. Gelinas, M. Tagmouti, and B. Jaumard, "Data models for service failure prediction in supply-chain networks," *arXiv preprint* arXiv:1810.09944, 2018, doi: 10.48550/arXiv.1810.09944.
[11] S. Kapoor and A. Narayanan, "Leakage and the reproducibility crisis in machine-learning-based science," *Patterns*, vol. 4, no. 9, p. 100804, 2023, doi: 10.1016/j.patter.2023.100804.
[12] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," in *Adv. Neural Inf. Process. Syst. 30 (NIPS 2017)*, 2017, pp. 3146–3154. [Online]. Available: https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf
[13] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953.
[14] R. van den Goorbergh, M. van Smeden, D. Timmerman, and B. Van Calster, "The harm of class imbalance corrections for risk prediction models: illustration and simulation using logistic regression," *J. Am. Med. Inform. Assoc.*, vol. 29, no. 9, pp. 1525–1534, 2022, doi: 10.1093/jamia/ocac093.
[15] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," *PLoS ONE*, vol. 10, no. 3, p. e0118432, 2015, doi: 10.1371/journal.pone.0118432.
[16] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Adv. Neural Inf. Process. Syst. 30 (NIPS 2017)*, 2017, pp. 4765–4774, doi: 10.48550/arXiv.1705.07874.
[17] B. Van Calster, D. J. McLernon, M. van Smeden, L. Wynants, and E. W. Steyerberg, "Calibration: the Achilles heel of predictive analytics," *BMC Med.*, vol. 17, Art. no. 230, 2019, doi: 10.1186/s12916-019-1466-7.
[18] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, Art. no. 44, 2014, doi: 10.1145/2523813.
[19] L. J. Tashman, "Out-of-sample tests of forecasting accuracy: an analysis and review," *Int. J. Forecast.*, vol. 16, no. 4, pp. 437–450, 2000, doi: 10.1016/S0169-2070(00)00065-0.

Qurrotul Aini: *Corresponding Author
Copyright © 2027, Qurrotul Aini, Mohammad Al-Habsie, Muhammad Rafi Habibie, Nia Kumaladewi.