Journal Computer and Technology
Vol. 4, No. 1, July 2026, pp. 29-47
ISSN: 3048-1880 (Online)
DOI: https://doi.org/10.69916/comtechno.v4i1.514

Original Research

A Leakage-Aware Ensemble Framework for Imbalanced Tabular Data: Mitigating SMOTE Contamination in Hotel Cancellation Prediction

Gibran Maulana Syamroni[1], Bahtiar Imran[2], Surni Erniwati*[3], Zaeniah[4], Wenti Ayu Wahyuni[5]

[1,3] Department of Informatics Engineering, Faculty of Computer Science and Artificial Intelligence, Mataram University of Technology, Mataram, Indonesian
[2] Department of Computer Engineering, Faculty of Computer Science and Artificial Intelligence, Mataram University of Technology, Mataram, Indonesian
[4,5] Information System, Faculty of Computer Science and Artificial Intelligence, Mataram University of Technology, Mataram, Indonesian
Email: [1]gibransyamroni65@gmail.com, [2]bahtiarimranlombok@gmail.com, [3]mentari1990@gmail.com, [4]zaen1989@gmail.com, [5]wentiayu443322@gmail.com

ARTICLE HISTORY
Received: May 29, 2026
Revised: July 13, 2026
Published: July 21, 2026

ABSTRACT

Predictive modeling on large-scale, imbalanced tabular data is frequently compromised by target leakage and improper resampling, leading to inflated performance metrics. While tree-based ensemble methods like Random Forest (RF) and XGBoost are widely deployed, their architectural divergence in handling complex behavioral anomalies under strict class imbalance remains underexplored. This study proposes a leakage-aware ensemble framework to mitigate SMOTE contamination and target leakage in hotel cancellation prediction. Using a rigorous CRISP-DM pipeline on 119,390 records, we applied SMOTE exclusively to the training set and engineered six behavioral features to capture non-linear contradictions, such as the counter-intuitive 99.36% cancellation rate in non-refund deposits. We systematically benchmarked RF (bagging) against XGBoost (boosting) using stratified 5-fold cross-validation, hyperparameter optimization, and loss curve monitoring. Results demonstrate that XGBoost structurally outperforms RF in minority-class detection, achieving superior Recall (0.6659), F1-Score (0.6607), and AUC-ROC (0.8657), with significantly lower variance (0.0035). Conversely, RF exhibited higher Precision (0.6588) and better cross-validation stability during hyperparameter search. Crucially, feature importance analysis revealed a structural divergence: RF prioritized temporal variables (lead_time), while XGBoost emphasized behavioral commitment signals (parking, special requests). These findings confirm that gradient boosting's sequential residual-correction mechanism is inherently more robust than variance-reduction bagging for imbalanced tabular data containing complex, non-linear anomalies. The proposed leakage-free framework not only resolves methodological flaws in prior studies but also provides a reliable, proactive risk-scoring foundation for integrating real-time decision support systems in production environments.

Keywords: Ensemble Learning; Data Leakage; SMOTE; Machine Learning; Tabul Data.

How to Cite:
G. M. Syamroni, B. Imran, S. Erniwati, Zaeniah, and W. A. Wahyuni, "A Leakage-Aware Ensemble Framework for Imbalanced Tabular Data: Mitigating SMOTE Contamination in Hotel Cancellation Prediction," *J. Comput. Technol.*, vol. 4, no. 1, pp. 29–47, 2026.

1. INTRODUCTION

Hotel reservation cancellations are a persistent and costly problem for the hospitality industry. When a guest cancels without sufficient notice, the room cannot realistically be filled on short notice, and the revenue from that night disappears permanently. This is not a marginal issue. Studies consistently show cancellation rates between 20% and 45% across property types, with online booking platforms making last-minute cancellations easier than ever [1],[2]. The result is what this research calls occupancy disruption: a gap between anticipated and realized room occupancy that directly damages financial performance.

The economic stakes are significant. Hotels operating near capacity can absorb some cancellations through overbooking strategies, but those strategies carry their own risks, including reputation damage when confirmed guests are turned away [3]. A more reliable solution is accurate early prediction. If a hotel system can flag high-risk reservations at the moment of booking, management has options: targeted retention outreach, adjusted deposit policies, or calibrated overbooking that accounts for predicted cancellations rather than historical averages [4].

Machine learning has produced increasingly accurate cancellation prediction models over the past several years. Early work focused on logistic regression and decision trees. More recent studies moved toward ensemble methods, with Random Forest and XGBoost emerging as the most consistently high-performing approaches on tabular reservation data [5], [6], [7], [8], [9]. However, many published comparisons either include only one of these two algorithms or evaluate them alongside fundamentally different model types (neural networks, Naive Bayes) without isolating the specific differences between bagging and boosting approaches [10].

A direct head-to-head comparison between Random Forest and XGBoost on this specific problem is less common than one might expect. Luo [11] achieved strong results with XGBoost at 98.48% accuracy and Random Forest at 94.89% AUC on a hotel dataset, but did not explore why one outperformed the other in terms of architectural differences. Solang and Adu [12] found Random Forest outperforming XGBoost in their multi-model evaluation and explicitly called for further investigation. Herrera et al. [9] demonstrated effective hyperparameter tuning for neural networks on hotel cancellation data without directly engaging the XGBoost alternative. Liu [13] compared multiple classifiers and found Random Forest achieving the highest accuracy, but excluded XGBoost from the comparison entirely.

The research gap is clear: there is no study that directly compares the bagging approach of Random Forest against the boosting approach of XGBoost on hotel cancellation data while also analyzing training stability through loss curve monitoring, applying systematic feature engineering, and using SMOTE to handle class imbalance properly. The presence of class imbalance in cancellation datasets is well-documented, but handling it correctly during cross-validated hyperparameter search requires careful design to avoid data leakage [14],[15]. This study addresses these gaps directly. Using the publicly available Hotel Booking Demand dataset of 119.390 records, we implement a full CRISP-DM pipeline that covers systematic preprocessing, feature engineering of six new variables, SMOTE applied exclusively to the training set, and hyperparameter optimization for both algorithms. The resulting comparison reveals not only which algorithm performs better overall, but why, through confusion matrix decomposition, loss curve analysis, and feature importance comparison across both models. The practical output is a validated prediction model suitable for integration into hotel Property Management Systems as a proactive cancellation risk scorer [16].

2. RESEARCH METHODS

2.1. Research Framework

This study adopts the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework, which structures the research into six iterative phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. The framework is particularly appropriate here because the research problem involves a production-grade prediction task where business requirements directly shape preprocessing and evaluation decisions [17], [18].

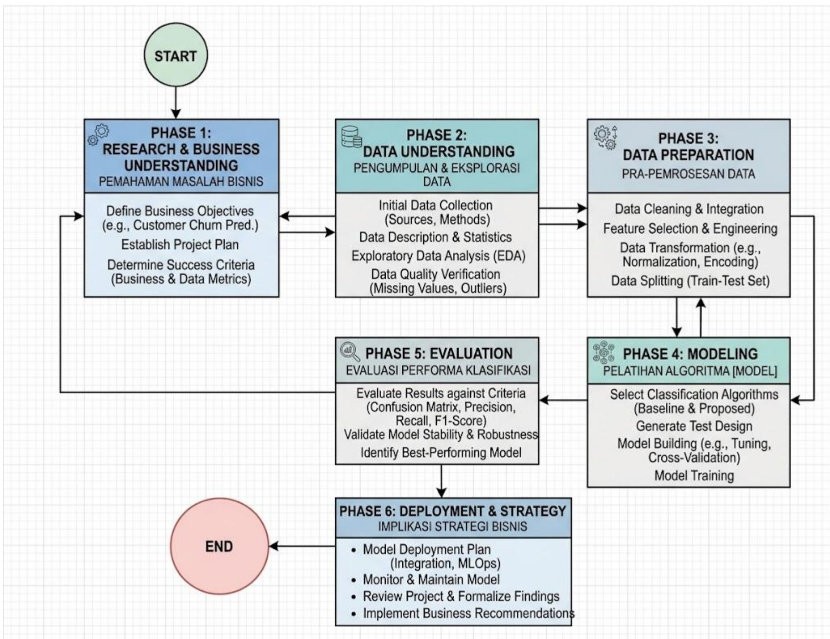

Figure 1. CRISP-DM research framework

2.2. Dataset

The Hotel Booking Demand dataset used in this study was obtained from Kaggle (https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand). The dataset was originally introduced by Antonio, de Almeida, and Nunes (2019), who collected the data from the Property Management System (PMS) records of two hotels located in Portugal. It contains comprehensive information on hotel reservations spanning the period from July 1, 2015, to August 31, 2017 [19], including both resort hotel and city hotel bookings. The dataset comprises a wide range of booking-related attributes, such as customer demographics, reservation details, stay duration, booking modifications, distribution channels, market segments, payment information, special requests, and booking status. Owing to its diversity of features and real-world booking records, the Hotel Booking Demand dataset has become a widely adopted benchmark for developing and evaluating machine learning models for hotel booking cancellation prediction and related hospitality analytics tasks. It contains 119.390 rows and 32 columns, mixing categorical and numerical features. The target variable is_canceled takes value 1 from cancelled reservation (44.224 records, 37.04%) and 0 for completed stays (75.166 records, 62.96%). This class imbalance justified the use of SMOTE during training.

Table 1. Predictor and Target Variables

Variables	Columns	Data Type
Predictor Variables	31 Columns	Numeric & Categorical
Target	1 Columns	Integer (0 & 1)

2.3. Data Preprocessing

Missing values appeared in four columns: children (4 values, imputed with mode), country (488 values, replaced with 'Unknown'), agent (16.340 values, replaced with 'No Agent"), and company (112.593 values, representing 94.31% values, of rows, replaced with 'No Company'). These categorical replacements preserve the distinction between genuinely missing agency information and confirmed direct bookings [4]. Deduplication removed 31.994 rows (26.80% of original data). The columns reservation_status and reservation_status_date were dropped entirely because they reveal the outcome after the fact, which could constitute data leakage in any realistic deployment scenario [14]. Outliers in the ADR (Average Daily Rate) column with values exceeding 1.000 were removed based on domain knowledge.

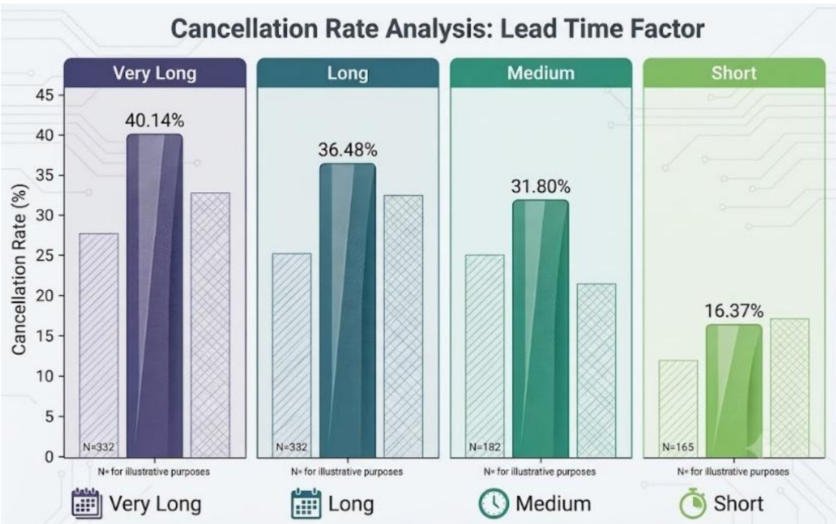

Figure 2. Cancellation Rate by Lead Time Category

As illustrated in Figure 2, the feature engineering process generated six new variables: total_stay_nights (the sum of weekend and weekday nights), total_guests (the total number of adults, children, and babies), is_family (a binary indicator of the presence of at least one child or baby), has_special_request (a binary indicator indicating whether any special request was made), is_same_room_type (indicating whether the reserved room type matched the assigned room type), and lead_time_category (an ordinal feature with four groups: Short [0–30 days], Medium [31–90 days], Long [91–365 days], and Very Long [over 365 days]). Following feature engineering, records with zero guests were removed during the data cleaning stage, resulting in a final dataset containing 87,196 booking records [7]. All categorical variables were converted to numeric using Label Encoding. For lead_time_category, manual ordinal mapping (Short=0, Medium=1, Long=2, Very Long=3) preserved the logical ordering. Data splitting used an 80:20 train-test ratio with stratified sampling (stratify=y) to maintain the 37.04% cancellation proportion in both subsets, yielding 69.756 training records and 17.440 test records. SMOTE (Small Minority Over-Sampling Technique) was applied exclusively to the training set after splitting [20], [21], increasing training data to 87.916 balanced samples. Applying SMOTE before splitting would contaminate the test set and inflate reported performance metrics [14].

2.4. Algorithms

a. Random Forest. Random Forest is a bagging ensemble that constructs multiple decision trees on bootstrapped subsets of the training data and aggregates predictions via majority voting [5], [22], [23], [24]. The final prediction y is:

$$\hat{y} = \mathrm{mode}\{h_1(x), h_2(x), \dots, h_n(x)\} \qquad (1)$$

where $\hat{y}$ denotes the final class prediction, $x$represents the input feature vector, $h_i(x)$is the prediction of the i-th decision tree, $n$is the total number of trees in the Random Forest ensemble, and $\mathrm{mode}(\cdot)$denotes the majority voting function used to determine the final predicted class.

Node splits are determined by minimizing Gini Impurity:

$$Gini(D) = 1 - \sum_{i=1}^{c} p_i^2 \qquad (2)$$

where $\mathrm{Gini(D)}$denotes the Gini impurity of dataset D, Drepresents the dataset at the current decision node, $\mathrm{p_i}$is the proportion of samples belonging to class i, and cis the total number of classes. A smaller Gini impurity value indicates a purer node and is therefore preferred when selecting the optimal split during decision tree construction.

Feature selection at each split uses Gini Gain:

$$\Delta\mathrm{Gini} = \mathrm{Gini(D)} - \sum_{\mathrm{j}=1}^{\mathrm{k}} \frac{|\mathrm{D_j}|}{|\mathrm{D}|}\mathrm{Gini}(\mathrm{D_j}) \qquad (3)$$

where $\Delta Gini$ denotes the reduction in Gini impurity after a dataset split, $Gini(D)$is the impurity of the parent dataset D, $D_j$represents the j-th child subset generated by the split, $k$is the total number of child subsets, $|\ D\ |$and $|\ D_j\ |$denote the numbers of samples in the parent and child datasets, respectively, and $\frac{|D_j|}{|D|}$ represents the proportion of samples in each child subset. The split with the highest $\Delta Gini$is selected because it produces the greatest reduction in node impurity. Bagging reduces variance by averaging predictions across independently trained trees, which is why Random Forest is resistant to overfitting [12].

b. XGBoost (Extreme Gradient Boosting). XGBoost builds trees sequentially, where each new tree corrects the residual errors of the current ensemble [9]. It minimizes an objective function combining a loss term and a regularization penalty:

$$Obj(\theta) = \sum_{i=1}^{n} l(\hat{y}_i, y_i) + \sum_{k=1}^{K} \Omega(f_k) \qquad (4)$$

where $Obj(\theta)$denotes the objective function of XGBoost, θrepresents the model parameters, $n$is the number of training samples, $l(\hat{y}_i, y_i)$is the loss function between the predicted value $\hat{y}_i$and the true label y_i, $K$is the total number of trees in the ensemble, $f_k$denotes the k-th decision tree, and $\Omega(f_k)$ is the regularization function that penalizes model complexity to improve generalization and prevent overfitting.

For binary classification, the logistic loss function is:

$$L(y_i, p_i) = -[y_i ln(p_i) + (1 - y_i) ln(1 - p_i)] \qquad (5)$$

where $L(y_i, p_i)$ denotes the binary cross-entropy loss for the i-th training sample, $y_i$is the true binary class label, $p_i$is the predicted probability of the positive class, and $\ln(\cdot)$represents the natural logarithm. The loss function penalizes incorrect probability predictions, and lower loss values indicate better agreement between the predicted probabilities and the true class labels.

Overfitting is controlled by a regularization term:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \qquad (6)$$

2.5. Modeling

Random Forest (RF) operates via bagging: it builds multiple decision trees independently on random feature and sample subsets, the aggregates predictions by majority vote. This parallel construction produces stable, low-variance predictions but treats each tree as an independent learner with no correction of previous errors [5]. XGBoost operates via gradient boosting: it builds trees sequentially, with each new tree focusing on correcting the residual errors of the previous ensemble. This sequential error correction makes XGBoost particularly effective on imbalanced data where correctly identifying the minority class requires iterative refinement [6], [9] . Baseline models for both algorithms used n_estimators=100 and random_state=42. RF optimization used RandomizedSearchCV with 3-fold cross-validation, searching: n_estimators in {100, 200}, max_depth in {10, 20, None}, min_samples_split in {2, 5, 10}, and min_samples_leaf in {1, 2, 4}. XGBoost optimization used GridSearchCV over: n_estimators in {100, 200}, max_depth in {3, 5, 7}, learning_rate in {0.1, 0.01}, subsample in {0.8, 1.0}, and colsample_bytree in {0.8, 1.0}. F1-Score served as the optimization target for both searches .

2.6. Evaluation Metrics

Model performance was assessed using five metrics derived from the confusion matrix. Accuracy measures the fraction of total prediction that are correct. Precision measures what fraction of predicted cancellations are actual cancellations. Recall (sensitivity) measures what fraction of actual cancellations the model detected. F1-Score is the harmonic mean of Precision and Recall, balancing both. AUC-ROC measures the model's ability to discriminate between classes across all classification thresholds [22].

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \qquad (7)$$

$$\text{Precision} = \frac{TP}{TP + FP} \qquad (8)$$

$$\text{Recall} = \frac{TP}{TP + FN} \qquad (9)$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (10)$$

For hotel operations, Recall carries the highest practical weight. A False Negative (a genuine cancellation the model missed) means the hotel takes no preventive action and loses the revenue entirely. A False Positive (a predicted cancellation that did not occur) wastes some retention resources but does not cause a room to sit empty [3]. Stratified 5-fold cross-validation was used to assess stability across data partitions. XGBoost training stability was additionaly evaluated via loss curve analysis (log loss per boosting round for both training and validation/test sets).

3. RESULTS AND DISCUSSION

3.1. Exploratory Data Analysis

EDA on the Hotel Booking Demand dataset confirmed the class imbalance: 44.224 canceled reservations (37.04%) versus 75.166 check-ins (62.96%). This gap justifies both the use of ensemble algorithms and SMOTE.

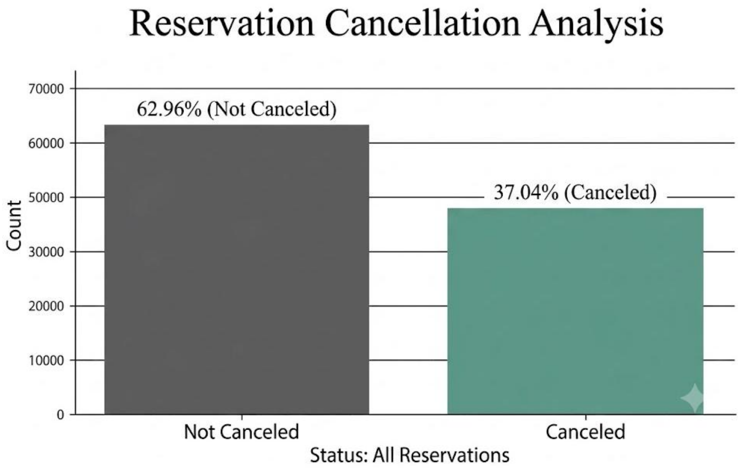

Figure 3. Reservation Cancellation Distribution

Figure 3 illustrates the distribution of reservation statuses in the dataset, categorized into Not Canceled and Canceled reservations. The results indicate that the majority of reservations were not canceled, accounting for 62.96% of the total records, while 37.04% of reservations were canceled. This distribution reveals a moderate class imbalance, with non-canceled reservations occurring more frequently than canceled ones. The observed imbalance suggests that customers generally tend to honor their reservations, although a substantial proportion of bookings are still canceled. Since more than one-third of all reservations result in cancellation, accurately predicting cancellation behavior is essential for improving hotel occupancy management, revenue optimization, and operational planning. Furthermore, the class imbalance should be considered during model development and evaluation to ensure that predictive models do not become biased toward the majority class and maintain reliable performance in identifying canceled reservations.

The most striking EDA finding came from the deposit_type variable. Reservations under the Non Refund policy canceled at a 99.36% rate, the opposite of what financial logic would suggest. Guests apparently cancel non-refundable bookings at near-total rates, which implies that external pressures (travel emergencies, speculative OTA bookings) outweigh the financial penalty of losing a deposit. This gives deposit_type very strong predictive signal for both algorithms.

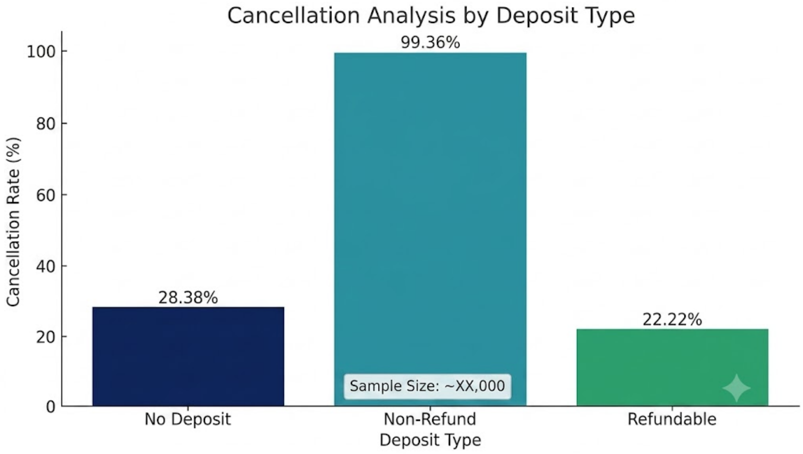

Figure 4. Cancellation Rate by Deposit Type

Figure 4 presents the hotel reservation cancellation rate for each deposit type. The results show that reservations with a Non Refund deposit have the highest cancellation rate, reaching 99.36%, indicating that almost all bookings under this deposit category were ultimately canceled. In contrast, reservations with a No Deposit policy have a cancellation rate of 28.38%, while those with a Refundable deposit exhibit the lowest cancellation rate at 22.22%. These findings suggest that deposit type is a highly influential factor in reservation cancellation behavior. The exceptionally high cancellation rate associated with the Non Refund category may reflect specific booking conditions, customer behavior, or operational policies within the dataset rather than the expected effect of non-refundable payments. Consequently, deposit type should be considered an important predictive feature in machine learning models for hotel reservation cancellation. Understanding this relationship enables hotel managers to design more effective booking policies, optimize revenue management strategies, and reduce the financial impact of reservation cancellations.

From the market segment angle, the Groups segment canceled at nearly 62%, the highest of any segment followed by OTA 36.72%, consistent with the speculative booking patterns inherent to online intermediaries [1].

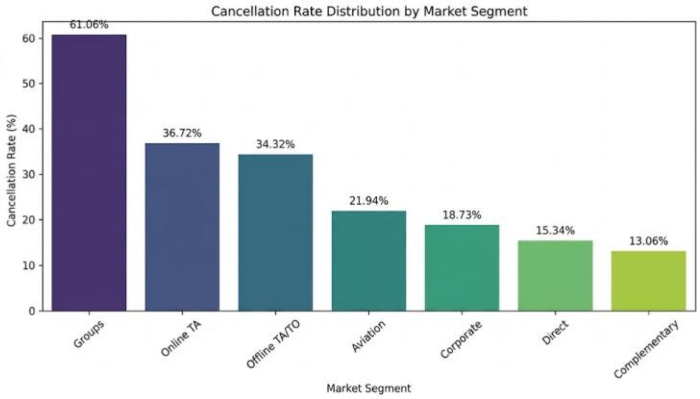

Figure 5. Cancellation Rate by Market Segment

Figure 5 illustrates the reservation cancellation rate across different market segments. Among all segments, the Groups market segment exhibits the highest cancellation rate at 61.66%, followed by Online TA with 36.72% and Offline TA/TO with 34.32%. In comparison, Aviation and Corporate bookings have moderate cancellation rates of 21.94% and 18.73%, respectively, while the Direct and Complementary segments show the lowest cancellation rates at 15.34% and 13.06%. These results indicate that market segment has a significant influence on reservation cancellation behavior. Reservations made through group bookings and travel agencies are considerably more likely to be canceled than those made directly with the hotel or through complementary arrangements. This pattern may be attributed to the greater flexibility, bulk booking practices, or itinerary changes commonly associated with group and agency reservations. Consequently, market segment represents an important predictive feature for machine learning models aimed at forecasting hotel reservation cancellations. Incorporating this variable can help improve prediction accuracy and assist hotel managers in developing targeted cancellation mitigation strategies for high-risk customer segments.

3.2. Random Forest Results

The baseline Random Forest (n_estimators=100, random_state=42) achieved Accuracy=0.8110, Precision=0.6558, Recall=0.6404, F1-Score=0.6493, and AUC-ROC=0.8548. After RandomizedSearchCV tuning (best configuration: n_estimators=100, max_depth=None, min_sample_split=5, min_samples_leaf=2; best CV score=0.8273), all five metrics improved. Table 2 summarizes the comparison:

Table 2. Baseline vs Tuned Random Forest

No	Metrics	RF Baseline	RF Tuned	Increased Score
1	Accuracy	0.8110	0.8132	+0.0022
2	Precision	0.6582	0.6588	+0.0006
3	Recall	0.6406	0.6549	+0.0143
4	F1-Score	0.6493	0.6569	+0.0076
5	AUC-ROC	0.8548	0.8626	+0.0078

Table 2 compares the performance of the baseline Random Forest model with the tuned Random Forest model after hyperparameter optimization. The results demonstrate that hyperparameter tuning consistently improves the model across all evaluation metrics. The overall accuracy increased from 0.8110 to 0.8132, representing a modest improvement of 0.0022, indicating a slight enhancement in the model's overall classification performance. More substantial improvements are observed in the classification metrics for the positive class. Precision increased slightly from 0.6582 to 0.6588 (+0.0006), indicating that the proportion of correctly predicted reservation cancellations remained relatively stable. In contrast, recall improved from 0.6406 to 0.6549 (+0.0143), demonstrating that the tuned model successfully identified a larger proportion of actual canceled reservations. Consequently, the F1-score, which balances precision and recall, increased from 0.6493 to 0.6569 (+0.0076), reflecting a more balanced classification performance. Furthermore, the AUC-ROC value increased from 0.8548 to 0.8626 (+0.0078), indicating that the tuned

Random Forest model achieved better discrimination between canceled and non-canceled reservations across different decision thresholds. Although some improvements are relatively small, the consistent increase across all evaluation metrics confirms that hyperparameter tuning enhanced the predictive capability and generalization performance of the Random Forest model. These results demonstrate that optimizing model parameters can produce a more reliable classifier for hotel reservation cancellation prediction without introducing overfitting.

The most operationally meaningful improvement was in Recall (+0.0143), meaning the tuned model detects meaningfully more actual cancellations. AUC-ROC also improved substantially (+0.0078), indicating better probabilistic discrimination across thresholds. That every metric moved in the positive direction without any tradeoff confirms that RandomizedSearchCV efficiently explored the parameter space, a result consistent with Azhar et al. [12] who showed that hyperparameter optimization consistently lifts Random Forest performance on hotel data.

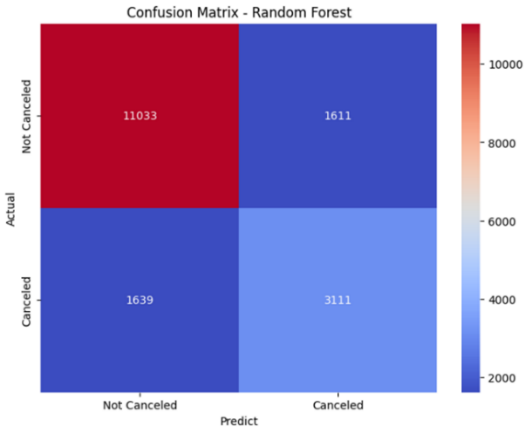

Figure 6. Confusion Matrix of the Tuned Random Forest Model

Figure 6 presents the confusion matrix of the tuned Random Forest model for hotel reservation cancellation prediction. The confusion matrix summarizes the model's classification performance by comparing the actual reservation status with the predicted outcomes. The model correctly classified 11,033 non-canceled reservations as Not Canceled (True Negatives) and successfully identified 3,111 canceled reservations as Canceled (True Positives). However, the model incorrectly predicted 1,611 non-canceled reservations as canceled (False Positives) and failed to identify 1,639 canceled reservations, classifying them as non-canceled (False Negatives).

These results indicate that the Random Forest model performs well in distinguishing between canceled and non-canceled reservations, with a notably high number of correct predictions in both classes. The relatively balanced numbers of false positives and false negatives suggest that the model does not exhibit a strong bias toward either class. Moreover, the high number of true positives contributes to the model's improved recall, while the relatively low number of false positives supports a stable precision value. Overall, the confusion matrix confirms that the tuned Random Forest model provides reliable classification performance, making it suitable for assisting hotel managers in identifying reservations that are likely to be canceled and enabling more effective operational and revenue management decisions.

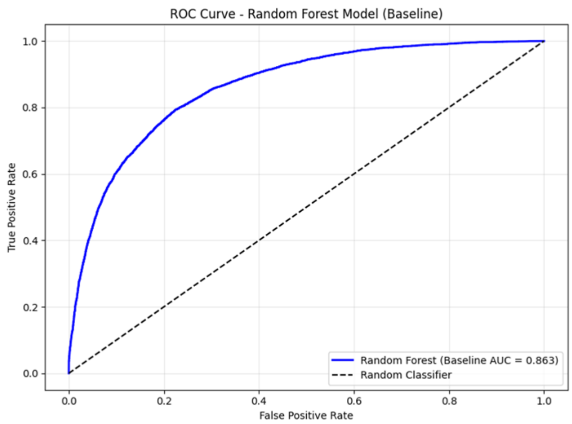

Figure 7. Receiver Operating Characteristic (ROC) Curve of the Baseline Random Forest Model

Figure 7 illustrates the Receiver Operating Characteristic (ROC) curve of the baseline Random Forest model for hotel reservation cancellation prediction. The ROC curve evaluates the model's ability to distinguish between canceled and non-canceled reservations by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various classification thresholds. The dashed diagonal line represents the performance of a random classifier, while the blue ROC curve represents the predictive performance of the Random Forest model. The model achieved an Area Under the ROC Curve (AUC-ROC) of approximately 0.863, indicating excellent discriminative capability. Since the ROC curve consistently lies well above the diagonal reference line, the model performs substantially better than random guessing across all decision thresholds. An AUC value above 0.80 generally reflects strong classification performance, demonstrating that the Random Forest model is highly effective in distinguishing reservations that will be canceled from those that will not. Furthermore, the steep rise of the ROC curve near the origin indicates that the model achieves a high true positive rate while maintaining a relatively low false positive rate. This characteristic is desirable for hotel reservation cancellation prediction because it enables the identification of a large proportion of canceled reservations without generating excessive false alarms. Overall, the ROC analysis confirms that the baseline Random Forest model provides reliable predictive performance and establishes a strong foundation for further improvement through hyperparameter tuning.

3.3. XGBoost Results

The baseline XGBoost (n_estimators=100, eval_metrics='logloss') performed well without any tuning: Accuracy=0.8113, Precision=0.6521, Recall=0.6621, F1-Score=0.6571, AUC-ROC=0.8630. Compared to the Random Forest baseline, XGBoost already led on Recall (+0.0215), F1-Score (+0.0078), and AUC-ROC (+0.0082). The performance gap at the baseline level is not coincidental—XGBoost's sequential boosting mechanism naturally concentrates correction effort on misclassified minority-class instances, making it structually better suited for imbalanced cancellation data [9].

Table 3. Baseline vs Tuned XGBoost

No	Metrics	XGBoost Baseline	XGBoost Tuned	Increased Score
1	Accuracy	0.8113	0.8133	+0.0020
2	Precision	0.6521	0.6557	+0.0036
3	Recall	0.6621	0.6659	+0.0038
4	F1-Score	0.6571	0.6607	+0.0036
5	AUC-ROC	0.8630	0.8657	+0.0027

Table 3 presents a comparison between the baseline XGBoost model and the tuned XGBoost model after hyperparameter optimization. The results indicate that tuning led to consistent improvements across all evaluation metrics, demonstrating enhanced predictive performance. The accuracy increased from 0.8113 to 0.8133, representing an improvement of 0.0020, which indicates a slight increase in the model's overall classification correctness. The optimization process also improved the model's ability to identify canceled reservations. Precision increased from 0.6521 to 0.6557 (+0.0036), indicating that a greater proportion of reservations predicted as canceled were actually canceled. Similarly, recall improved from 0.6621 to 0.6659 (+0.0038), showing that the tuned model was able to detect more actual cancellation cases. As a result, the F1-score, which balances precision and recall, increased from 0.6571 to 0.6607 (+0.0036), reflecting a more effective and balanced classification performance. In addition, the AUC-ROC value increased from 0.8630 to 0.8657 (+0.0027), indicating a stronger ability of the tuned model to distinguish between canceled and non-canceled reservations across various classification thresholds. Although the improvements are relatively modest, the consistent enhancement across all metrics confirms that hyperparameter tuning successfully improved the generalization capability and predictive accuracy of the XGBoost model.

Overall, the results demonstrate that the tuned XGBoost model outperforms the baseline configuration and provides a more reliable approach for hotel reservation cancellation prediction. The improvements in precision, recall, F1-score, and AUC-ROC suggest that the optimized model can better support hotel management in anticipating potential cancellations and making data-driven operational decisions.

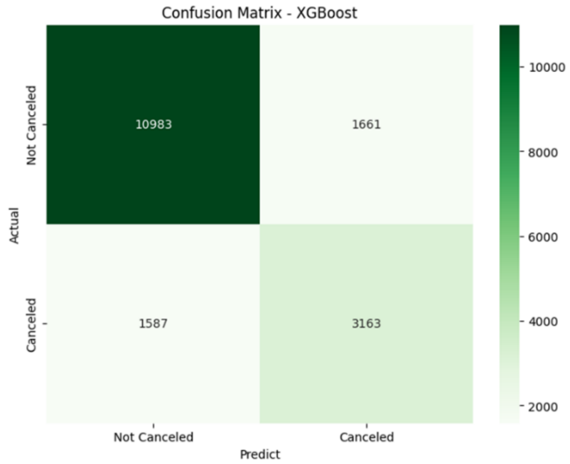

Figure 8. Confusion Matrix of the Tuned XGBoost Model

Figure 8 presents the confusion matrix of the tuned XGBoost model for hotel reservation cancellation prediction. The confusion matrix compares the actual reservation status with the model's predicted outcomes, providing a detailed view of the classification performance. The model correctly classified 10,983 non-canceled reservations as Not Canceled (True Negatives) and accurately identified 3,163 canceled reservations as Canceled (True Positives). However, 1,661 non-canceled reservations were incorrectly predicted as canceled (False Positives), while 1,587 canceled reservations were incorrectly classified as non-canceled (False Negatives). The confusion matrix indicates that the tuned XGBoost model achieves strong classification performance for both reservation classes. Compared with the Random Forest model, XGBoost correctly identifies a slightly higher number of canceled reservations, as reflected by the larger number of true positives and fewer false negatives. This contributes to the model's improved recall and F1-score, enabling it to detect reservation cancellations more effectively while maintaining competitive precision.

Overall, the confusion matrix demonstrates that the tuned XGBoost model provides reliable and balanced predictions for hotel reservation cancellation. Its ability to correctly classify a large proportion of both canceled and non-canceled reservations makes it an effective tool for supporting hotel managers in forecasting cancellations, optimizing room allocation, and implementing proactive revenue management strategies.

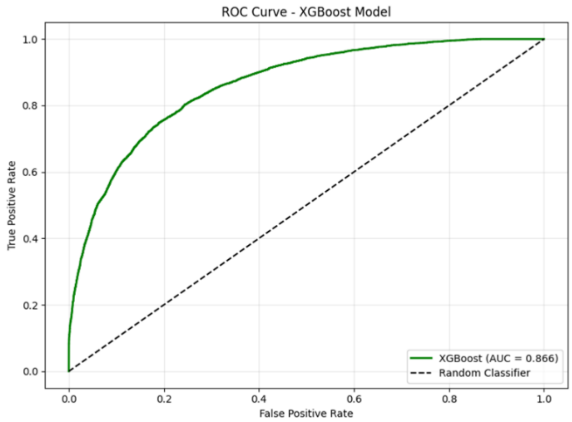

Figure 9. Receiver Operating Characteristic (ROC) Curve of the Tuned XGBoost Model

Figure 9 presents the Receiver Operating Characteristic (ROC) curve of the tuned XGBoost model for hotel reservation cancellation prediction. The ROC curve evaluates the model's classification performance by illustrating the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different classification thresholds. The dashed diagonal line represents the performance of a random classifier, while the green ROC curve demonstrates the predictive capability of the tuned XGBoost model. The tuned XGBoost model achieved an Area Under the ROC Curve (AUC-ROC) of approximately 0.866, indicating excellent discriminative performance. The ROC curve remains well above the diagonal reference line throughout the entire threshold range, demonstrating that the model consistently distinguishes canceled reservations from non-canceled reservations more effectively than random classification. An AUC value greater than 0.85 suggests that the model has a high probability of correctly ranking a randomly selected canceled reservation higher than a randomly selected non-canceled reservation.

Furthermore, the steep increase of the ROC curve at low false positive rates indicates that the tuned XGBoost model is capable of identifying a large proportion of canceled reservations while maintaining relatively few false alarms. This balance between sensitivity and specificity makes the model particularly suitable for hotel reservation cancellation prediction, where accurately identifying potential cancellations is essential for effective occupancy planning and revenue management. Overall, the ROC analysis confirms that the tuned XGBoost model provides robust and reliable predictive performance and slightly outperforms the tuned Random Forest model in terms of discriminative ability, as reflected by its higher AUC-ROC value.

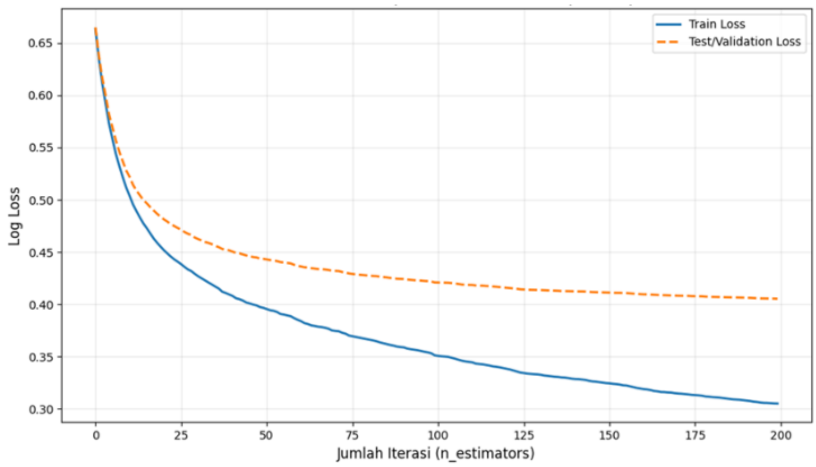

Figure 10. Training and Validation Loss Curve of the Tuned XGBoost Model

Figure 10 illustrates the training and validation loss curves of the tuned XGBoost model during the optimization process. The horizontal axis represents the number of boosting iterations (n_estimators), while the vertical axis shows the logarithmic loss (Log Loss) for both the training and validation datasets. The solid blue line corresponds to the training loss, whereas the dashed orange line represents the validation loss. As the number of boosting iterations increases, both the training and validation loss decrease steadily, indicating that the model progressively improves its predictive performance. The training loss decreases from approximately 0.66 to 0.31, while the validation loss declines from approximately 0.66 to 0.41. The rapid reduction in loss during the early iterations demonstrates that the model quickly learns the underlying patterns in the data, whereas the slower decline in later iterations indicates convergence toward an optimal solution.

Although a noticeable gap exists between the training and validation loss curves, the validation loss continues to decrease gradually without showing a significant increase. This behavior suggests that the tuned XGBoost model does not suffer from severe overfitting and maintains good generalization performance on unseen data. The stable convergence of both curves confirms that the selected hyperparameters provide an effective balance between learning the training data and preserving predictive capability for new observations. Consequently, the loss curve supports the evaluation results presented in previous experiments, demonstrating that the tuned XGBoost model is robust and suitable for hotel reservation cancellation prediction.

3.4. Feature Importance Analysis

Feature importance rangkings differed between the two algorithms, reflecting their different internal mechanisms. In Random Forest, lead_time ranked first, followed by adr and has_special_request. This reflects Random Forest's Gini Impurity-based importance metric, which tends to favor high-cardinality continuous variables that provide many candidate split points. In XGBoost, required_car_parking_spaces ranked first, followed by has_special_request and deposit_type. XGBoost measures feature importance by frequency of use in residual-correcting trees, which means categorical features with strong discriminative power, like parking need (a binary proxy for guest seriousness), score disproportionately high [15].

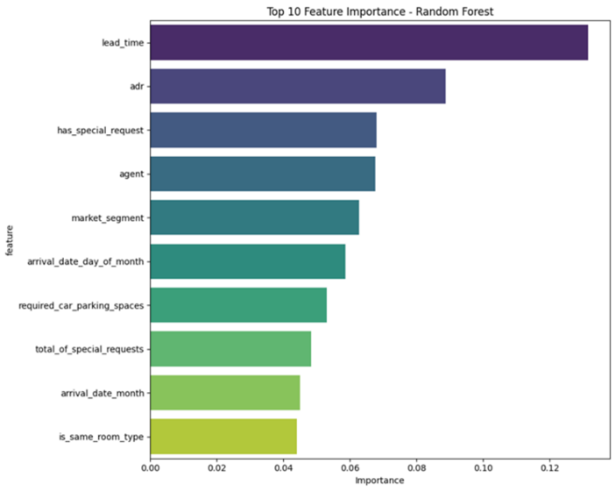

Figure 11. Top 10 Feature Importance of the Random Forest Model

Figure 11 presents the top ten most important features identified by the tuned Random Forest model for predicting hotel reservation cancellations. Feature importance reflects the relative contribution of each variable to the model's decision-making process. Among all features, lead_time has the highest importance score, making it the most influential predictor of reservation cancellation. This is followed by adr (Average Daily Rate), has_special_request, agent, and market_segment, which also contribute substantially to the model's predictive performance. The remaining important features include arrival_date_day_of_month, required_car_parking_spaces, total_of_special_requests, arrival_date_month, and is_same_room_type. Although these variables have lower importance scores than the top-ranked features, they still provide valuable information that enhances the model's ability to distinguish between canceled and non-canceled reservations. The dominance of lead_time suggests that the time interval between the booking date and the arrival date is the strongest indicator of cancellation behavior. Reservations made well in advance are generally more susceptible to changes in customer plans, resulting in a higher likelihood of cancellation. Similarly, the high importance of adr indicates that room pricing significantly influences customer decisions, while variables related to booking characteristics and customer preferences, such as special requests, market segment, and booking agent, further improve prediction accuracy. Overall, the feature importance analysis demonstrates that the Random Forest model effectively captures the relative influence of multiple reservation attributes, providing valuable insights into the key factors associated with hotel reservation cancellations.

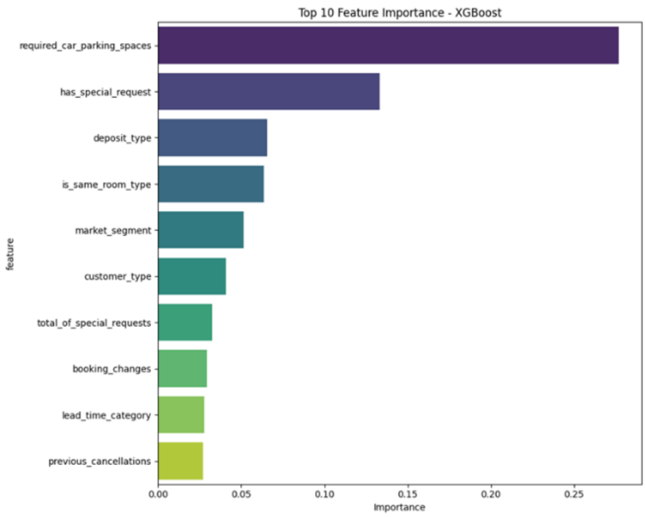

Figure 12. Top 10 Feature Importance of the XGBoost Model

Figure 12 presents the top ten most important features identified by the tuned XGBoost model for predicting hotel reservation cancellations. Feature importance represents the contribution of each variable to the model’s prediction process. Among all variables, required_car_parking_spaces has the highest importance score by a substantial margin, indicating that parking space requirements are the most influential factor in determining whether a reservation is likely to be canceled. The second most important feature is has_special_request, followed by deposit_type and is_same_room_type, which also make significant contributions to the model's predictive performance.

Other influential features include market_segment, customer_type, total_of_special_requests, booking_changes, lead_time_category, and previous_cancellations. Although these variables contribute less than the leading features, they collectively provide valuable information that enhances the model's ability to accurately classify reservation outcomes. The feature importance distribution indicates that the XGBoost model captures a different set of influential predictors compared with the Random Forest model. While Random Forest identified lead_time as the dominant feature, XGBoost places considerably greater emphasis on required_car_parking_spaces, suggesting that the gradient boosting algorithm detects stronger predictive relationships between parking requirements and reservation cancellation behavior. Likewise, booking characteristics such as deposit type, market segment, customer type, and customers' historical behavior, represented by previous_cancellations and booking_changes, play meaningful roles in the prediction process. Overall, the feature importance analysis demonstrates that the tuned XGBoost model effectively prioritizes the variables that contribute most to reservation cancellation prediction, providing valuable insights into the key factors influencing customer booking decisions. The four consistent predictors collectively represent a guest commitment dimension: guests who make special requests, need parking, and operate in committed market segments are far less likely to cancel. These findings align with Silvestre et al. [5] who identified lead_time and deposit_type as primary cancellation drivers; the current study adds parking and special request signals as equally robust across algorithm paradigms.

3.5. Model Comparison and Cross-Validation

To provide a comprehensive evaluation of the proposed approaches, the performance of the tuned Random Forest and tuned XGBoost models was compared using several standard classification metrics. The comparison includes accuracy, precision, recall, F1-score, AUC-ROC, and cross-validation (CV) F1-score to assess not only predictive performance on the test dataset but also the generalization capability of each model. The results of this comparison are presented in Table 4.

Table 4. Model Comparison Random Forest vs XGBoost

No	Metrics	Random Forest	XGBoost	Superior
1	Accuracy	0,8132	0,8133	XGBoost
2	Precision	0,6588	0,6557	Random Forest
3	Recall	0,6549	0,6659	XGBoost
4	F1-Score	0,6569	0,6607	XGBoost
5	AUC-ROC	0,8626	0,8657	XGBoost
6	CV F1-Score	0,8273	0,8164	Random Forest

Table 4 demonstrates that both machine learning models achieve highly competitive performance in predicting hotel reservation cancellations. The tuned XGBoost model slightly outperforms the tuned Random Forest model in most predictive metrics, achieving the highest accuracy (0.8133), recall (0.6659), F1-score (0.6607), and AUC-ROC (0.8657). The higher recall indicates that XGBoost is more effective at correctly identifying canceled reservations, while its superior F1-score reflects a better balance between precision and recall. Moreover, the highest AUC-ROC value confirms that XGBoost provides stronger discriminative capability across different classification thresholds. In contrast, the tuned Random Forest model achieves slightly better precision (0.6588) and a higher cross-validation F1-score (0.8273) than XGBoost. The higher precision suggests that Random Forest generates fewer false positive predictions, while the stronger cross-validation performance indicates better stability and generalization across different data partitions. These results imply that Random Forest produces more consistent performance when evaluated using repeated validation procedures.

XGBoost leads on Accuracy (0.8133 vs. 0.8132), Recall (0.6659 vs. 0.6549), F1-Score (0.6607 vs. 0.6569), and AUC-ROC (0.8657 vs. 0.8626). Random Forest holds advantages in Precision (0.6588 vs. 0.6557) and cross-validation F1-Score (0.8273 vs. 0.8164). These results reflect a genuine architectural difference rather than noise: XGBoost's sequential boosting optimizes for detection coverage (Recall) at a slight cost to precision, while Random Forest's parallel aggregation yields more conservative, higher-precision predictions [8].

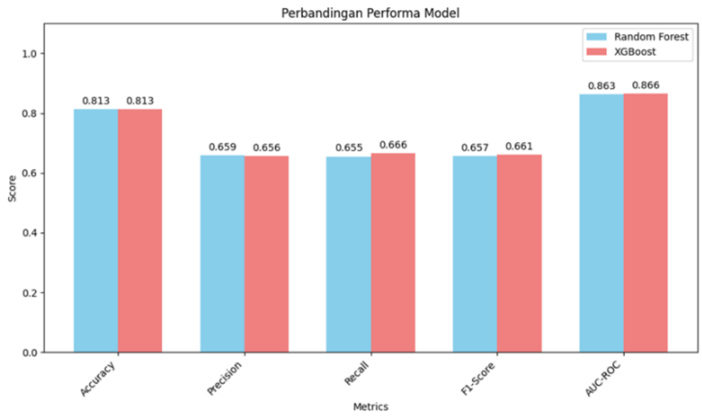

Figure 13. Performance Comparison between the Tuned Random Forest and XGBoost Models

Figure 13 compares the predictive performance of the tuned Random Forest and tuned XGBoost models across five evaluation metrics: Accuracy, Precision, Recall, F1-score, and AUC-ROC. The bar chart illustrates that both models achieve very similar overall performance, with only slight differences across the evaluation metrics. The comparison provides a clear visual representation of the strengths of each model in predicting hotel reservation cancellations. The results show that XGBoost achieves marginally higher accuracy (0.8133) than Random Forest (0.8132), indicating a slightly better overall classification performance. XGBoost also outperforms Random Forest in recall (0.6659 vs. 0.6549), F1-score (0.6607 vs. 0.6569), and AUC-ROC (0.8657 vs. 0.8626). These improvements suggest that XGBoost is more effective at identifying canceled reservations while maintaining a better balance between precision and recall. In particular, the higher AUC-ROC demonstrates a stronger ability to distinguish canceled from non-canceled reservations across different classification thresholds.

On the other hand, Random Forest achieves a slightly higher precision (0.6588) compared to XGBoost (0.6557), indicating that it produces marginally fewer false positive predictions when classifying reservation cancellations. Although this advantage is relatively small, it reflects the conservative nature of the Random Forest classifier in predicting the positive class.

The ROC curves for both models sit well above the random-chance diagonal, with AUC values of 0.863 (RF) and 0.866 (XGBoost), a difference that is small in absolute terms but consistent with the directional advantage seen across other metrics.

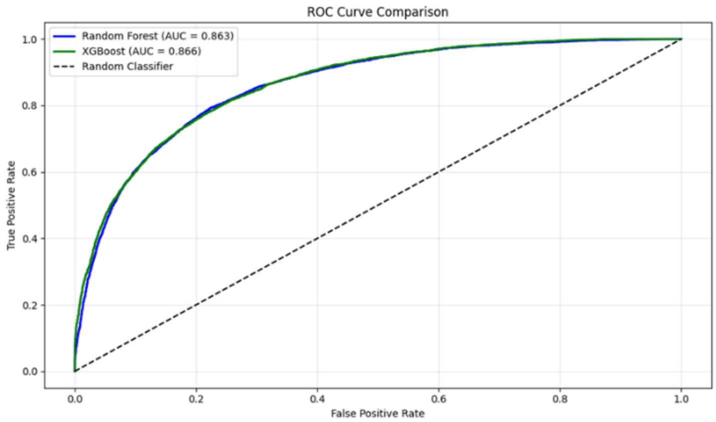

Figure 14. Receiver Operating Characteristic (ROC) Curve Comparison between the Random Forest and XGBoost Models

Figure 14 compares the Receiver Operating Characteristic (ROC) curves of the tuned Random Forest and XGBoost models for hotel reservation cancellation prediction. The ROC curve illustrates the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different classification thresholds, providing a comprehensive assessment of each model's discriminative ability. The dashed diagonal line represents the performance of a random classifier, while the blue and green curves correspond to the Random Forest and XGBoost models, respectively. The comparison shows that both models achieve excellent classification performance, with ROC curves positioned well above the random classification line. The Random Forest model obtains an AUC-ROC of

0.863, whereas the XGBoost model achieves a slightly higher AUC-ROC of 0.866. Although the numerical difference is relatively small, the XGBoost ROC curve remains marginally above the Random Forest curve across most classification thresholds, indicating a consistently stronger ability to distinguish between canceled and non-canceled reservations.

The similarity of the two ROC curves demonstrates that both ensemble learning algorithms are highly effective for hotel reservation cancellation prediction. However, the slightly higher AUC-ROC achieved by XGBoost suggests that it offers better overall discriminative performance and greater robustness when classifying reservations under varying decision thresholds. This superior performance is consistent with the evaluation metrics presented in the previous experiments, where XGBoost achieved higher accuracy, recall, F1-score, and AUC-ROC than Random Forest. Overall, Figure 14 confirms that both models provide reliable predictive capabilities for hotel reservation cancellation. Nevertheless, XGBoost emerges as the preferred model due to its consistently superior ROC performance and greater ability to correctly separate canceled reservations from non-canceled reservations, making it a more suitable choice for practical applications in hotel occupancy forecasting and revenue management.

To further evaluate the robustness and generalization capability of the proposed models, a Stratified 5-Fold Cross-Validation experiment was conducted. Stratified cross-validation preserves the original class distribution in each fold, ensuring that both canceled and non-canceled reservations are proportionally represented throughout the validation process. The F1-score was selected as the primary evaluation metric because it provides a balanced assessment of precision and recall, making it particularly suitable for datasets with moderately imbalanced class distributions. The cross-validation results for the tuned Random Forest and XGBoost models are summarized in Table 5.

Table 5. Stratified 5-Fold Cross-Validation Performance Comparison of Random Forest and XGBoost

No	Fold	F1-Score RF	F1-Score XGB	Superior
1	Fold 1	0.6335	0.6388	XGB
2	Fold 2	0.6303	0.6328	XGB
3	Fold 3	0.6179	0.6303	XGB
4	Fold 4	0.6338	0.6389	XGB
5	Fold 5	0.6294	0.6347	XGB
-	Mean	0.6290	0.6351	XGB
-	Std. Deviation	0.0065	0.0035	XGB

Table 5 presents the results of the Stratified 5-Fold Cross-Validation for both the tuned Random Forest and XGBoost models. Across all five validation folds, XGBoost consistently achieved a higher F1-score than Random Forest, demonstrating superior predictive performance and greater consistency during the cross-validation process. The F1-scores obtained by XGBoost ranged from 0.6303 to 0.6389, while the Random Forest model produced F1-scores ranging from 0.6179 to 0.6338. The average F1-score further confirms this advantage, with XGBoost achieving a mean score of 0.6351, compared with 0.6290 for Random Forest. Although the difference in the average F1-score is relatively modest, the consistent superiority of XGBoost across all validation folds indicates that the gradient boosting approach learns more discriminative patterns from the reservation data and maintains better predictive performance on unseen samples. In addition to achieving a higher mean F1-score, XGBoost also exhibits a lower standard deviation (0.0035) compared with Random Forest (0.0065). The smaller standard deviation indicates that XGBoost produces more stable and consistent results across different training and validation splits, suggesting stronger generalization capability and lower sensitivity to variations in the dataset. This stability is an important characteristic for real-world hotel reservation cancellation prediction, where models are expected to maintain reliable performance when applied to new booking data.

Overall, the cross-validation results reinforce the findings of the previous experiments. While both ensemble learning models demonstrate strong predictive capability, XGBoost consistently outperforms Random Forest in terms of both predictive accuracy and stability. The combination of a higher mean F1-score and lower performance variability indicates that XGBoost is the more robust model for hotel reservation cancellation prediction and is therefore recommended as the final predictive model in this study.

3.6. Managerial Implications

The XGBoost model's Recall advantage translates into a measurable operational benefit. At 10.000 reservations, it detects roughly 110 more cancellations than Random Forest. Each detected cancellations is an action trigger: the hotel can launch a personalized follow-up, offer a retention incentive (upgrade, discount), or open the room to the waitlist before the rate window closes. Groups and OTA guests without special requests, without parking needs, with long lead times, and with prior cancellation history represent the highest-risk profiles, the natural targets for automated intervention.

Integration into a Property Management System (PMS) via a prediction API would allow risk scores to be computed for every incoming reservation and surfaced in the hotel's operational dashboard. This positions the model not as a reporting tool but as a real-time decision support layer, precisely the kind of proactive infrastructure that Afolorunso et al. [3] argue is necessary to replace reactive overbooking strategies with evidence-based ones. Random Forest, while narrowly trailing on aggregate metrics, retains a clear role: its higher Precision (0.6588 vs. 0.6557) and stronger CV F1-Score make it a better fit for contexts where false alarms are operationally costly, such as non-refundable upgrade offers that the hotel cannot retract if the guest actually shows up. The two models are thus complementary rather than simply competitive.

3.7. Discussion

The findings of this study demonstrate that ensemble learning methods are highly effective for predicting hotel reservation cancellations. Both Random Forest and XGBoost achieved strong classification performance, with accuracy values exceeding 81% and AUC-ROC values above 0.86. These results indicate that both models are capable of distinguishing between canceled and non-canceled reservations with high reliability. The relatively high AUC values also suggest that the models maintain good discriminative ability across different classification thresholds, making them suitable for practical decision-support systems in hotel revenue management. The experimental results further reveal that hyperparameter tuning contributes positively to model performance. Although the improvements in accuracy, precision, recall, F1-score, and AUC-ROC are relatively modest, both Random Forest and XGBoost consistently benefited from parameter optimization. This finding highlights the importance of selecting appropriate hyperparameter configurations rather than relying solely on default settings. Optimized parameters enable the models to better capture complex relationships within reservation data while maintaining strong generalization performance.

A comparison between the two algorithms indicates that XGBoost slightly outperformed Random Forest in most evaluation metrics, including accuracy, recall, F1-score, and AUC-ROC. The higher recall achieved by XGBoost is particularly important in the context of hotel reservation cancellation prediction because it enables the identification of a larger proportion of reservations that are likely to be canceled. Early identification of these reservations allows hotel managers to implement proactive strategies such as dynamic pricing, overbooking control, promotional campaigns, or customer reminders to minimize revenue loss caused by unexpected cancellations. Although Random Forest achieved marginally higher precision, the overall superiority of XGBoost demonstrates the effectiveness of gradient boosting in modeling complex nonlinear relationships among reservation attributes.

The confusion matrix analysis provides additional insight into the predictive behavior of both models. Both classifiers correctly identified the majority of canceled and non-canceled reservations while maintaining relatively balanced numbers of false positives and false negatives. This balance indicates that neither model is excessively biased toward the majority class despite the moderate class imbalance observed in the dataset. Furthermore, the ROC curve comparison confirms that XGBoost consistently provides slightly better discrimination capability than Random Forest, supporting the quantitative evaluation results obtained from the classification metrics.

The feature importance analysis offers valuable practical insights into the factors influencing hotel reservation cancellations. The Random Forest model identified lead_time as the most influential predictor, suggesting that reservations made further in advance are more susceptible to cancellation due to changes in customer travel plans. In contrast, XGBoost assigned the highest importance to required_car_parking_spaces, indicating that customer service preferences and booking characteristics also play significant roles in predicting cancellation behavior. Other influential variables, including deposit_type, market_segment, average daily rate (ADR), special requests, and booking changes, further demonstrate that reservation cancellation is determined by multiple interacting factors rather than by a single variable. These findings are consistent with previous studies that identify booking lead time, customer behavior, and reservation characteristics as key determinants of hotel booking cancellations. The Stratified 5-Fold Cross-Validation results further strengthen the reliability of the proposed models. XGBoost consistently achieved higher F1-scores across all validation folds while exhibiting a lower standard deviation than Random Forest. This indicates that XGBoost not only provides superior predictive performance but also demonstrates greater stability and robustness when applied to different subsets of the data. Such consistency is essential for real-world deployment, where predictive models must maintain reliable performance under varying reservation patterns and operational conditions.

From a managerial perspective, the proposed prediction models can provide significant benefits for hotel management. Accurate cancellation prediction enables hotels to optimize occupancy planning, implement more effective overbooking strategies, improve revenue forecasting, and allocate resources more efficiently. By identifying reservations with a high probability of cancellation before the scheduled arrival date, hotel managers can take preventive actions to reduce financial losses and improve room utilization. Consequently, integrating machine learning–based prediction models into hotel reservation management systems can support more informed and data-driven decision-making.

Despite the promising results, this study has several limitations. The models were developed and evaluated using a single publicly available hotel reservation dataset, which may not fully represent booking patterns across different geographical regions, hotel categories, or customer segments. Additionally, the study focused exclusively on structured reservation data without incorporating external factors such as weather conditions, public holidays, local events, economic indicators, or online customer sentiment, all of which may influence cancellation behavior. Future research could investigate the integration of these external variables, explore deep learning and hybrid ensemble approaches, or develop explainable artificial intelligence (XAI) techniques to improve both predictive performance and model interpretability.

4. CONCLUSION

This study built and compared Random Forest and XGBoost for hotel reservation cancellation prediction under controlled experimental conditions on the Hotel Booking Demand dataset (119.390 entries, 37.04% cancellation rate). After systematic preprocessing, including deduplication of 31.994 records, six-variable feature engineering, and SMOTE based class balancing on the training set, the tuned XGBoost model delivered the best aggregate performance: Accuracy of 0.8133, Recall of 0.6659, F1-Score of 0.6607, and AUC-ROC of 0.8657. These results beat the tuned Random Forest on four of five metrics, with the Recall gap (+0.0110) representing ~110 additional cancellations detected per 10.000 reservations at the operational level. Cross-validation confirmed XGBoost's edge: a mean 5-Fold F1-Score of 0.6351 with standard deviation of 0.0035, compared to 0.6290 and 0.0065 for Random Forest. The lower variance makes XGBoost more reliable for deployment across varied booking patterns. Loss curve analysis showed no overfitting, validation loss stabilized without rising throughout 200 boosting iterations. Random Forest retained two specific advantages: higher Precision (0.6588) and a better cross-validation F1-Score during hyperparameter search (0.8273), making it the preferred choice when false alarms carry significant operational cost. Four features proved consistently predictive across both algorithms, has_special_request, market_segment, total_of_special_requests, and required_car_parking_spaces, confirming that guest commitment signals outperform purely temporal variables as universal cancellation indicators. The Non Refund deposit type, despite its financial disincentive for cancellation, showed a 99.36% cancellation rate in EDA, a finding that challenges conventional deposit policy logic and deserves dedicated follow-up research. Practically, the XGBoost model is ready for integration into hotel PMS infrastructure as a proactive decision-support tool, flagging high-risk bookings for targeted intervention. Future work should explore LightGBM and CatBoost as additional ensemble candidates, incorporate external signals (competitor pricing, weather, local events), and test on datasets from Indonesian domestic hotels to assess geographic generalizability. Explainable AI techniques (SHAP values) would also strengthen managerial trust in model-driven decisions.

DECLARATIONS

Author Contributions

G.M.S. and B.I. conceived the study and designed the methodology. G.M.S. performed the data collection and preprocessing. B.I. and S.E. conducted the data analysis, modeling, and interpretation of results. Z. and W.A.W. contributed to the literature review, manuscript editing, and validation. All authors discussed the results and contributed to the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Conflicts of Interest

The authors declare no conflict of interest regarding the publication of this paper.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

AI Use Statement

During the preparation of this work, the authors used ChatGPT and Google Gemini in order to improve the language and readability of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the final content of the published article.

Additional Information

No additional information is available for this paper.

REFERENCES

[1] E. Rahmawati and G. S. Nurohim, “Optimization of Prediction for Cancellation of Hotel Room Reservation Using Decision Tree with Feature Selection and Resampling,” *J. Sist. Inf. Bisnis*, vol. 15, no. 2, pp. 211–215, 2025, doi: 10.14710/vol15iss2pp211-215.

[2] Z. Zafitri and M. I. Jambak, “Karakteristik Pembatalan Reservasi Kamar Hotel Pada Online Travel Agent Menggunakan Algoritma C4.5,” *Indones. J. Comput. Sci.*, vol. 12, no. 4, pp. 2010–2023, 2023, doi: 10.33022/ijcs.v12i4.3268.

[3] A. A. Afolorunso, B. E. Aimuel, A. O. Abiodun, A. O. Adesina, and S. A. Ajagbe, “Development of a Machine Learning–Enabled Decision Support Framework for Hotel Booking Cancellation Prediction,” *Adv. Multidiscip. Sci. Res. J. Publ.*, vol. 16, no. 3, pp. 23–36, 2025, doi: 10.22624/aims/cisdi/v16n3p2.

[4] Y. Azhar, G. A. Mahesa, and M. C. Mustaqim, “Prediksi pembatalan pemesanan hotel menggunakan optimalisasi hiperparameter pada algoritme Random Forest,” *J. Teknol. dan Sist. Komput.*, vol. 9, no. 1, pp. 15–21, 2021, doi: 10.14710/jtsiskom.2020.13790.

[5] P. Silvestre, N. Antonio, and P. Carrasco, “Navigating uncertainty: enhancing hotel cancellation predictions with adaptive machine learning,” *Inf. Technol. Tour.*, vol. 28, no. 1, 2026, doi: 10.1007/s40558-025-00349-9.

[6] Saifudin, “Komparasi Akurasi Metode Random Forest Dan Deep Learning pada Reservasi Hotel,” *IMTechno J. Ind. Manag. Technol.*, vol. 4, no. 2, pp. 124–128, 2023, doi: 10.31294/imtechno.v4i2.2026.

[7] Y. Zhao and P. Qin, “Hotel Booking Cancellation Prediction by Feature Interaction and Machine Learning,” 2026, pp. 315–320. doi: 10.1145/3779475.3779521.

[8] B. K. and B. Dwarakanath, “Hybrid model for detection of brain tumor using convolution neural networks,” *Comput. Sci. Inf. Technol.*, vol. 5, no. 1, pp. 84–90, 2024, doi: 10.11591/csit.v5i1.pp84-90.

[9] A. Herrera, Á. Arroyo, A. Jiménez, and Á. Herrero, “Forecasting hotel cancellations through machine learning,” *Expert Syst.*, vol. 41, no. 9, pp. 1–19, 2024, doi: 10.1111/exsy.13608.

[10] D. Hartanti, A. Ichsan Pradana, and S. Lestari, “Komparasi Algoritma Decision Tree, SVM dan ANN untuk Reservasi Hotel,” *Duta.com : Jurnal Ilmiah Teknologi Informasi dan Komunikasi*, vol. 16, pp. 21–27, 2023, doi: 10.47701/dutacom.v16i1.2647

[11] Z. Luo, *Hotel Cancellation Rate Prediction: A Machine Learning Based Prediction Model*, no. Iciaai. Atlantis Press International BV, 2025. doi: 10.2991/978-94-6463-823-3_31.

[12] E. W. Solang and F. X. Adu, “Machine Learning Evaluation for Hotel Cancellation Prediction with Threshold Adjustment and Cost-Based Evaluation Evaluasi Machine Learning untuk Prediksi Pembatalan Hotel dengan Threshold Adjustment dan Cost-Based Evaluation,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science,* vol. 6, no. January, pp. 193–204, 2026, doi: 10.57152/malcom.v6i1.2466.

[13] Y. Liu, “Research for Hotel Reservation Cancellation Based on Prediction Model,” *Adv. Econ. Manag. Polit. Sci.*, vol. 153, no. 1, pp. 154–162, 2024, doi: 10.54254/2754-1169/2024.19518.

[14] Andy Hermawan, Aji Saputra, Nabila Lailinajma, Reska Julianti, Timothy Hartanto, and Troy Kornelius Daniel, “Predicting Hotel Booking Cancellations Using Machine Learning for Revenue Optimization,” *Router J. Tek. Inform. dan Terap.*, vol. 3, no. 1, pp. 37–48, 2025, doi: 10.62951/router.v3i1.400.

[15] E. Rahmawati and C. Agustina, “Perbandingan Teknik Resample pada Algoritma K-NN dan SVM untuk Prediksi Pembatalan Pemesanan Kamar Hotel,” *Jurnal Teknologi Informasi dan Terapan (J-TIT),* vol. 10, no. 2, pp. 102–107, 2023, doi: 10.25047/jtit.v10i2.333.

[16] F. H. Qani’ah, R. Ramadhan, A. C. Firdaus, and I. Veritawati, “Prediksi Pembatalan Reservasi Hotel Menggunakan Algoritma Naive Bayes,” *J. Informatics Adv. Comput.*, vol. 4, no. 1, pp. 76–80, 2023, doi: 10.35814/jiac.v4i1.5499

[17] S. Tripathi, D. Muhr, M. Brunner, H. Jodlbauer, M. Dehmer, and F. Emmert-Streib, “Ensuring the Robustness and Reliability of Data-Driven Knowledge Discovery Models in Production and Manufacturing.,” *Front. Artif. Intell.*, vol. 4, p. 576892, 2021, doi: 10.3389/frai.2021.576892.

[18] R. A. Casonatto, T. De Pádua Grillo Souza, and A. M. Mariano, “Quality and Risk Management in Data Mining: A CRISP-DM Perspective.,” *Procedia Comput. Sci.*, vol. 242, pp. 161–168, 2024, doi: 10.1016/j.procs.2024.08.257.

[19] N.-H. Anh-Khôi, L. Hà-Duy-Nguyên, and T. Vĩnh-Khang, “Artificial Intelligence Applied to Address Tourism Challenges: Predicting Hotel Room Cancellations,” in *Proceedings of the 11th International Conference on Emerging Challenges: Smart Business and Digital Economy 2023 (ICECH 2023)*, Atlantis Press International BV, 2023, pp. 434–445. doi: 10.2991/978-94-6463-348-1_33.

[20] N. S. Thomas and S. Kaliraj, “An Improved and Optimized Random Forest Based Approach to Predict the Software Faults,” *SN Comput. Sci.*, vol. 5, no. 5, 2024, doi: 10.1007/s42979-024-02764-x.

[21] K. Rajendran, M. Jayabalan, and V. Thiruchelvam, “Predicting breast cancer via supervised machine learning methods on class imbalanced data,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 8, pp. 54–63, 2020, doi:

10.14569/IJACSA.2020.0110808.
[22] B. Imran, H. Hambali, A. Subki, Z. Zaeniah, A. Yani, and M. R. Alfian, "Data Mining Using Random Forest, Naïve Bayes, and Adaboost Models for Prediction and Classification of Benign and Malignant Breast Cancer," *J. Pilar Nusa Mandiri*, vol. 18, no. 1, pp. 37–46, 2022, doi: 10.33480/pilar.v18i1.2912.
[23] B. Imran, E. Wahyudi, A. Subki, S. Salman, and A. Yani, "Classification of stroke patients using data mining with adaboost, decision tree and random forest models," *Ilk. J. Ilm.*, vol. 14, no. 3, pp. 218–228, 2022, doi: 10.33096/ilkom.v14i3.1328.218-228.
[24] V. Gupta, "The influencing role of social media in the consumer's hotel decision-making process," *Worldw. Hosp. Tour. Themes*, 2019, doi: 10.1108/WHATT-04-2019-0019.

.