Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/BA08.20250226
Published: December 20, 2025

Data-Driven Strategies to Combat Missed Appointments: Machine Learning Prediction and Optimization of Healthcare Resources

Md. Ashhab Rahman
Department of Industrial and Production Engineering
Military Institute of Science and Technology
Dhaka, Bangladesh
ashhabpro@gmail.com

Tasnimul Hasan Nishorgo
Department of Industrial and Production Engineering
Military Institute of Science and Technology
Dhaka, Bangladesh
nishorgo30@gmail.com

Tanjeel Ahmed Bin Zaman
Department of Industrial and Production Engineering
Military Institute of Science and Technology
Dhaka, Bangladesh
tanjeel@ipe.mist.ac.bd

Abstract

Missed medical appointments cause significant loss of time and resources of medical centers. Current systems mainly depend on automated or manual reminders to ensure attendance. In this scenario, introducing a machine learning model to predict no show of patients can significantly reduce missed appointments, increase revenue as well as allow for optimized scheduling for patients who need timely assessment. The main objective of this study was to predict patient no-shows using machine learning techniques for the hospitals to take data driven strategies. A dataset containing patient ID, appointment schedule, appointment date, disease type, and attendance status (show/no-show) was analyzed. We implemented and compared three modeling pipelines: a CatBoost-based resampled ensemble, a stacked LightGBM ensemble, and an AutoML pipeline using AutoGluon. Our best-performing approach, AutoGluon, achieved an accuracy of 0.808423053 and a ROC AUC of 0.763069012, producing well-calibrated probability estimates and outperforming traditional and ensemble baselines. Feature-importance analysis revealed that age, SMS reminder receipt, and scheduling-time features were among the strongest predictors of no-show behavior. These findings demonstrate that AutoGluon can effectively identify patients at high risk of missing appointments, offering a practical tool for healthcare providers to proactively intervene and optimize scheduling, potentially reducing missed visits, improving resource allocation, and enhancing patient care continuity.

Keywords
Predictive Modeling, Machine Learning, Appointment No-Shows, Healthcare Analytics and Patient Behavior.

1. Introduction

Appointment non-attendance, or patient no show is a common problem for healthcare institutions across the globe. Problems arise when a patient makes an appointment but fails to show up and does not provide notice. This situation creates gaps in care, causes delays in receiving care, and wastes medical resources. In the United States, patient no-shows are estimated to cost the healthcare system about $150 billion a year (*Can Machine Learning Predict Patient No-Shows? - AAPC Knowledge Center*, n.d.)

A study estimated that an incidence of 67,000 patient no-shows could result in an approximate cost of $7 million to the healthcare system. A survey indicated that independent physician practices can face an estimated annual loss of up to $150,000 due to missed appointments. (*How Much Each Year Do No Shows Cost the U.S. Healthcare System?*, n.d.; Marbouh et al., 2020)

Patients with chronic illnesses, diabetes, and hypertension also suffer adverse outcomes when appointments are missed and not followed up. No-show rates globally vary by healthcare settings and patient demographics, ranging from 3% to 80%. Research has identified multiple factors influencing appointment no-shows, including socioeconomic status, transportation barriers, scheduling conflicts, and health literacy. Research has identified various factors affecting appointment no-shows, including socioeconomic status, transportation obstacles, scheduling issues, and health literacy. Behavioral factors, including forgetfulness and a perceived lack of urgency for the appointment, greatly contribute as well. In Brazil, where public healthcare systems cater to extensive and varied populations, recorded no-show rates may exceed 30%, causing significant operational and financial burdens on local health services. (Kaplan-Lewis & Percac-Lima, 2013; Liu et al., 2022)

Traditionally little to no steps are taken to combat no-shows, however, SMS-alerts or reminder calls are sometimes used but that too becomes very ineffective in real-life scenarios. Recent studies have demonstrated that employing machine learning techniques to predict no-shows of patients can be useful. This study utilized a real-world dataset of medical appointments from Brazil, encompassing demographic, behavioral, and clinical features, including age, gender, appointment scheduling details, SMS reminders, chronic disease history, and scholarship status.

Advanced machine learning techniques, such as CatBoost-based Resampled Ensemble Classification, Stacked LightGBM Ensemble with Covariate-Dependent Stacking, and AutoGluon, were utilized to develop predictive models for patient no-shows. CatBoost proficiently handles categorical variables and imbalanced datasets, while AutoGluon facilitates automated model selection and hyperparameter tuning, hence reducing the complexity of model development. By comparing various techniques, we want to identify resilient and interpretable models that could aid healthcare professionals in improving appointment scheduling and resource allocation.

1.1 Objectives

This research seeks to forecast patient absences in outpatient sessions with sophisticated machine learning methodologies. The study's particular objectives are to uncover critical determinants affecting appointment attendance, encompassing demographic, behavioral, and clinical characteristics. The work aims to create and compare predictive models utilizing CatBoost-based Resampled Ensemble Classification, Stacked LightGBM Ensemble with Covariate-Dependent Stacking, and AutoGluon to attain precise and interpretable predictions. The study will assess model performance using rigorous criteria and identify the most effective strategy for practical implementation.

2. Literature Review

The studies on machine learning models across various works are listed in Table 1. The studies which were done on the Brazilian dataset are discussed afterwards.

Table 1. Studied machine learning models in previous related works

Reference	Models Evaluated	Best Model(s)
(Srinivas & Salah, 2021)	Stochastic Gradient Boosted Classification Tree with Deep Neural Network Regressor; Random Forest	Stochastic Gradient Boosted Classification Tree with Deep Neural Network Regressor
(Fan et al., 2021)	Random Forest; Logistic Regression; K-Nearest Neighbor; Gradient Boosting; Decision Tree; Bagging	Bagging

Reference	Models Evaluated	Best Model(s)
(Liu et al., 2022)	Logistic Regression; NaÏve Bayes; Artificial Neural Network	Artificial Neural Network
(Dunstan et al., 2023)	Random Forest; Logistic Regression; Gradient Boosting; AdaBoost; Support Vector Machine; RUS Boost; EasyEnsemble	—
(A. Alshammari et al., 2021)	Decision Tree; AdaBoost	Decision Tree
(Salazar et al., 2022)	Random Forest; Logistic Regression; Decision Tree	Random Forest
(Deina et al., 2024)	K-Nearest Neighbor; Support Vector Machine; Symbolic Regression	Symbolic Regression
(Batool et al., 2021)	K-Nearest Neighbor; Decision Tree; NaÏve Bayes; Support Vector Machine	Decision Tree
(Almeida et al., 2021)	Random Forest; Logistic Regression; Gradient Boosting; Artificial Neural Network	Logistic Regression
(Incze et al., 2021)	Logistic Regression; LightGBM	LightGBM
(Leiva-Araos et al., 2025)	Random Forest; Logistic Regression; XGBoost; Multi-Layer Perceptron	Random Forest
(Valero-Bover et al., 2022)	K-Nearest Neighbor; Decision Tree; Support Vector Machine; XGBoost	Decision Tree
(Osorio et al., 2025)	Random Forest; Gradient Boosting; Decision Tree; Support Vector Machine; Extra Trees; Bagging Random Forest With optimized hyperparameters by RandomForest; AdaBoost Decision Tree With optimized hyperparameters by DecisionTree	Bagging Random Forest With optimized hyperparameters by RandomForest
(R. Alshammari et al., 2020)	NaÏve Bayes; AdaBoost; Deep Neural Network	Deep Neural Network
(Barrera Ferro et al., 2020)	Random Forest; Logistic Regression; Artificial Neural Network	Artificial Neural Network
(Daghistani et al., 2020)	Random Forest; Logistic Regression; Gradient Boosting; Support Vector Machine; Artificial Neural Network	Gradient Boosting
(Zhang & Chen, 2025)	Random Forest; Logistic Regression; CatBoost	CatBoost
(Amalina et al., n.d.)	Random Forest; Logistic Regression; Decision Tree; NaÏve Bayes; MultiHead Attention Soft Random Forest	MultiHead Attention Soft Random Forest
(A. Alshammari et al., 2024)	Random Forest; Gradient Boosting; NaÏve Bayes; AdaBoost	Gradient Boosting
(Alaidah et al., 2021)	Random Forest; Logistic Regression; Decision Tree; XGBoost	Decision Tree
(Nguyen et al., 2023)	Logistic Regression; K-Nearest Neighbor; Gradient Boosting; Decision Tree; Artificial Neural Network; Gaussian NaÏve Bayes; Recurrent Neural Network	Recurrent Neural Network
(Abushaaban & Agaoglu, 2022)	Random Forest; Decision Tree; Support Vector Machine; Artificial Neural Network; Gaussian NaÏve Bayes	Random Forest

(A. Alshammari et al., 2021) studied the Brazilian dataset from the Kaggle database related to medical appointments, for appointments scheduled between April 29, 2016, and June 8, 2016 which included (110,528) appointments.

AdaBoost and Decision Tree algorithms were evaluated and contrasted based on multiple factors. AdaBoost surpassed Decision Tree in True Positive Rate (TPR), with a value of 0.95 versus 0.89. AdaBoost exhibited a False Negative Rate (FNR) of 0.17, whilst the Decision Tree demonstrated a value of 0.14. The precision for Decision Tree was marginally superior at 0.89, in contrast to AdaBoost's 0.87. Recall for the Decision Tree was 0.86, whilst AdaBoost attained 0.83. AdaBoost achieved a ROC (Receiver Operating Characteristic) score of 0.85, compared to 0.88 for the Decision Tree. Finally, AdaBoost achieved an F-measure of 0.84, somewhat inferior to Decision Tree's 0.87. The Decision Tree model demonstrated superior performance in predicting no-show compared to AdaBoost.

Four techniques of resampling were used in (Deina et al., 2024)'s study: Synthetic Minority Oversampling Technique (SMOTE), Random UnderSampling (RUS), NearMiss (NM), and Instance Hardness Threshold (IHT). KNN, SVM, and SR supervised algorithms were employed for predicting patient no-shows under each resampling technique. For Brazilian dataset's Test Portion, the best model was SR with IHT (Instance Hard Thresholding) resampling technique. It achieved the highest performance with an AUC of 0.9429 (SD: 0.048), Sensitivity of 0.9425 (SD: 0.049), Specificity of 0.9433 (SD: 0.046), and F1-Score of 0.9349 (SD: 0.065). The accuracy was 0.9429 (SD: 0.045). For the validation portion, the SR model with IHT again standed out with the best performance, showing an AUC of 0.9321 (SD: 0.037), Sensitivity of 0.9137 (SD: 0.070), Specificity of 0.9505 (SD: 0.037), and an F1-Score of 0.9349 (SD: 0.060). The accuracy was 0.9340 (SD: 0.035).

(Batool et al., 2021) intended to create a classification model to forecast hospital no-shows with enhanced accuracy compared to previous models using the Brazilian dataset. In their analysis, the Decision Tree (DT) attained the greatest accuracy of 94.5%, with Precision, Recall, and F1-measure each at 95%. The Naive Bayes (NB) classifier had the lowest performance, achieving an accuracy of 85%, a precision of 80%, a recall of 85%, and an F1-measure of 85%. The K-Nearest Neighbors (KNN) classifier achieved an accuracy of 92.5%, with Precision, Recall, and F1-measure all at 90.5%. The Support Vector Machine (SVM) utilizing a linear kernel attained an accuracy of 92%, with Precision at 89%, Recall at 88%, and an F1-measure of 88%. The SVM utilizing a nonlinear kernel exhibited performance comparable to that of the decision tree, achieving an accuracy of 94.5%, precision of 95%, recall of 95%, and an F1-measure of 95%. (Almeida et al., 2021) evaluated four algorithms on the identical dataset: Artificial Neural Network (ANN), Gradient Boosting (GB), Logistic Regression (LR), and Random Forest (RF). Logistic Regression had superior performance with a recall of 0.81, specificity of 0.79, and F1-score of 0.78, whereas Random Forest achieved an accuracy of 0.77, precision of 0.79, and F1-score of 0.77. ANN and GB demonstrated commendable performance, with ANN attaining a recall of 0.77 and GB reaching an accuracy of 0.76.

On the Brazilian dataset, (Zhang & Chen, 2025) integrated CatBoost with SMS reminders, reducing no-show rates from 32.1% to 18.5% and attaining an F1 score of 0.75 and a recall of 77%, with the most significant effect shown in younger patients and those without chronic diseases. (Alaidah et al., 2021) addressed class imbalance with a hybrid sampling (ALL K-NN + ADASYN), enabling Random Forest and XGBoost to achieve 0% false negatives and recall up to 100%, with SHAP confirming prior missed appointments and long lead times as key drivers. The summary of findings of the studies conducted on the Brazilian dataset are summarized in Table 2.

Table 2. Comparison of various machine learning algorithms used in previous studies.
(Here, x indicates the data was not found in the study)

Reference	Best Performing Model	Best Resampling Technique	Accuracy	ROC_AUC	Recall	F1_Score
(A. Alshammari et al., 2021)	Adaptive Boosting (AdaBoost)	x	x	0.85	0.83	0.84
(Deina et al., 2024)	Symbolic Regression (SR)	Instance Hard Thresholding (IHT)	0.6681	0.7734	0.9434	0.5214
(Batool et al., 2021)	Decision Tree (DT)	Instance Hard Thresholding (IHT)	0.945	x	0.95	0.95
(Almeida et al., 2021)	Logistic Regression (LR)	Synthetic Minority Oversampling Technique (SMOTE)	0.707	x	0.513	0.807
(Zhang & Chen, 2025)	Categorical Boosting (CatBoost)	Synthetic Minority Oversampling Technique (SMOTE)	x	0.64	0.77	0.75
(Alaidah et al., 2021)	Majority Voting Model of LR, DT and Random Forest (RF)	Hybrid ALL K-NN and Adaptive Synthetic Sampling (ADASYN)	0.865	x	0.9995	0.804

3. Methods

3.1 Data preprocessing and feature engineering

Different preprocessing pipelines were adopted for the three modelling processes. Common initial steps included: Mapping the target variable to a binary indicator (0/1) and excluding identifier columns and, in some cases, raw date columns to eliminate potential privacy identifiers or overfitting.

In the first (CatBoost-based) pipeline, categorical and numerical columns were differentiated by inspecting data types. All categorical variables (including integer-typed columns assumed to represent categorical data) were transformed with integer encoding via a label encoder, to facilitate subsequent sampling. Numerical features (e.g., age) were scaled using min–max normalization (Pereira et al., 2025) by equation 1.

$$Xscaled = \frac{(X - Xmin)}{(Xmax - Xmin)} \quad (1)$$

where X represents the original values of the feature, and Xmin and Xmax are its minimum and maximum observed values, respectively. After encoding and scaling, the dataset was split into training and test sets (stratified on the target) using an 80/20 split with a fixed random seed for reproducibility (random_state = 42).

Given the class imbalance typical in appointment-no-show data (i.e., fewer no-shows than shows), under-sampling technique was employed using the InstanceHardnessThreshold (IHT) method applied only to the training subset. This aims to reduce bias toward the majority class and make the classifier more sensitive to the minority class. The IHT method involves evaluating, for each majority-class instance, its "hardness," defined as the probability of being misclassified by a preliminary classifier (e.g., a Random Forest). Concretely, for a majority-class sample i with predicted class probability p_i of being majority, its hardness is expressed by equation 2.

$$hi = 1 - pi \tag{2}$$

Then, a threshold T is chosen so that only those majority-class instances whose hardness exceeds T (i.e., which the model predicts with relatively low confidence) are retained. If the desired final ratio of minority-to-majority classes is α_us, IHT seeks the percentile by equation 3.

$$T = \text{percentile}\{\, h_i \mid i \in \text{majority class}\,\} \text{ such that } \frac{N_m}{N_r M} = \alpha_{us} \tag{3}$$

where N_m is the number of minority-class instances and N_rM is the number of majority-class instances retained after resampling. This ensures that the resampled training set is balanced (or approximately so), while prioritizing "hard" majority instances likely near decision boundaries rather than random majority instances.

In the second (LightGBM stacking) pipeline, a more extensive feature engineering was performed using the raw date/time data. Specifically, scheduling and appointment timestamps were converted into a time difference variable (lead time in hours), as well as derived features such as the weekday (ScheduledDay_Weekday) and hour of scheduling (ScheduledDay_Hour). Categorical variables (e.g., neighborhood) were expanded via one-hot encoding (with drop-first to avoid multicollinearity). Missing values in numeric features (e.g., Age) were imputed using a simple mean imputation. Numeric features including age, lead-time, and scheduling-language features were standardized (zero mean, unit variance) using a standard scaler. As with Process 1, the data was split in a stratified 80/20 training/test split (random seed set to 42) and applied IHT under-sampling to the training set to rebalance classes. Hyperparameters of LightGBM such as learning rate (η), number of leaves (num_leaves), subsampling fraction (s), number of boosting rounds (T), and feature fraction (f) were tuned implicitly by selecting model variants that optimize the validation metric (e.g., ROC AUC). Thus, the LightGBM scoring function can be formalized by equation 4.

$$\hat{y} = \sum_{t=1}^{T} \eta \cdot h_t(X; \text{leaf parameters}) \tag{4}$$

with leaf parameters constrained by num_leaves, s, f, etc and hyperparameter tuning consists in selecting (T, η, num_leaves, s, f) that maximize cross-validated performance.

In the AutoML model, an automated modeling framework was employed via AutoGluon TabularPredictor (Erickson et al., 2020). After binary coding of the target and dropping identifier columns, the entire dataset was split into training and test partitions (80/20, stratified). Sample weights inversely proportional to class frequencies were computed (i.e., weighting minority class higher) to address class imbalance, rather than under-sampling or oversampling. These sample weights were provided to AutoGluon during model training. AutoGluon explores different hyperparameter configurations using strategies like random search, successive halving or Hyperband, within a given time budget. Formally, if H denotes the hyperparameter search space and M the set of model types, AutoGluon solves according to equation 5.

$$\text{argmax}_{m \in M,\, h \in H_m} \text{CV}_{\text{score}(m(X;h))} \tag{5}$$

subject to the resource constraints (e.g., time, CPU). The hyperparameter configuration that maximizes the cross-validated metric (e.g., ROC AUC) is then selected for final model training.

3.2 Model Development

Three unique categorization pipelines were developed and evaluated, each embodying a distinct modeling methodology. The initial pipeline, CatBoost-based Resampled Ensemble Classification, included three foundational learners: CatBoostClassifier, RandomForestClassifier, and LogisticRegression. Following the application of IHT-based resampling on the training set, three ensemble variants were created: (a) a soft VotingClassifier ensemble incorporating CatBoost, RandomForest, and Logistic Regression, (b) a bagging ensemble employing CatBoost (utilizing BaggingClassifier with CatBoost as the primary estimator), and (c) a stacking ensemble (StackingClassifier) comprising CatBoost and RandomForest as base learners, with Logistic Regression acting as the meta-learner. After training with the resampled data, each model was evaluated using the held-out test set, producing class predictions (binary: show vs. no-show) and estimated probabilities for each test sample.

The second pipeline, Stacked LightGBM Ensemble with Covariate-Dependent Stacking (Wakayama & Sugasawa, 2025), included three base models: LGBMClassifier (gradient boosting), RandomForestClassifier, and LogisticRegression (which includes class-weight balancing to address class imbalance). Five-fold stratified cross-validation was executed on the IHT-resampled training set to produce out-of-fold (OOF) probability predictions for each base model. Averaged base-model predictions were concurrently created for the test set. The base-model predictions, referred to as meta-features, were combined with a selection of covariate characteristics (including age, lead-time, and appointment scheduling temporal factors) to create the final meta-training and meta-test datasets. A meta-model, an additional LGBMClassifier, was subsequently trained on the meta-training set to generate the final predictions. To address class imbalance at the meta-model level, asymmetric class weights were employed, allocating a higher weight to the minority (no-show) class. A threshold sweep ranging from 0.1 to 0.9 was conducted for final classification, and the threshold that maximized the F1-score on the test data was selected. This strategy emphasized the improvement of the harmonic mean of precision and recall rather than simply focusing on accuracy.

The third pipeline, AutoGluon Tabular Pipeline, employed AutoGluon's tabular module, which automates several aspects of modeling, including model selection, hyperparameter tuning, and ensemble creation (Erickson et al., 2020). The sanitized dataset, from which IDs were excluded, underwent target encoding and the removal of ID columns, and was thereafter partitioned into training and testing subsets, with 20% allocated for testing purposes. Sample weights, inversely correlated with class frequency, were computed for the training data to address class imbalance and supplied to AutoGluon's "fit" algorithm to prioritize the minority (no-show) class during training. AutoGluon was executed using a "best_quality" preset with a time constraint (e.g., 4 hours) to achieve the optimal predictive model or ensemble within these limitations. Following training, predictions encompassing both class labels and probabilities were generated for the test set. Standard evaluation metrics were calculated, encompassing the confusion matrix, accuracy, sensitivity (recall), specificity, positive predictive value (PPV), negative predictive value (NPV), F1-score, area under the ROC curve (ROC AUC), area under the precision-recall curve (PR AUC), and the Brier score for probabilistic calibration. Furthermore, feature importance, as indicated by AutoGluon, was optionally derived.

3.3 Evaluation Metrics

For each model (or ensemble) across the three processes, predictive performance was evaluated on the test set never seen during training (or under-sampling) to provide an honest estimate of generalization. A standard battery of classification performance metrics was computed:

- **Confusion matrix**, leading to true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).
- **Accuracy**: proportion of correct predictions - $\frac{TP+TN}{TP+TN+FP+FN}$
- **Sensitivity (Recall for no-show class)**: $\frac{TP}{TP+FN}$
- **Specificity (True Negative Rate for show class)**: $\frac{TN}{TN+FP}$
- **F1-score**: the harmonic mean of precision and recall.
- **Area under the Receiver Operating Characteristic curve (ROC AUC)**, using predicted probabilities.

- **Area under the Precision–Recall curve (PR AUC)**, which is especially informative under class imbalance.

In the Stacked LightGBM Ensemble model, a threshold optimization step (based on F1) was also conducted to check whether a non-default decision threshold could improve performance, an approach sometimes employed in no-show prediction literature to balance sensitivity and precision under imbalanced outcomes.

Furthermore, for the AutoGluon model, feature-importance analyses and model-agnostic interpretability (e.g., SHAP analysis) were used to examine which predictors contributed most to no-show risk.

Computational environment

All experiments were implemented in Python 3.12. Hardware comprised an AMD Ryzen 5 3600 CPU, 16 GB DDR4-3200 RAM, and a GPU GTX 1660 Super, 6 GB. Random seeds (e.g., random_state=42) were set consistently to ensure reproducibility across splits and resampling.

Ethical considerations and reproducibility

Given that the dataset originates from historical scheduling records, identifiers were anonymized by removing patient and appointment IDs before modeling, ensuring that no individual patient could be re-identified in reported results. All preprocessing, resampling, modeling, and evaluation steps are described in full to support reproducibility. Where randomness is involved (data splitting, resampling, ensemble sampling), fixed seeds are used to allow replication. Following good reporting practices for biomedical data and ML workflows, detailed descriptions of data cleaning, encoding, resampling, model configuration, and evaluation were provided so that other researchers can reproduce or build upon this work. This transparency is consistent with the recommendations of methodological guidelines in biological machine learning research. (Seghier, 2022)

4. Data Collection

This study utilized a dataset sourced from Kaggle (*Medical Appointment No Shows*, n.d.) comprising 110,528 records of Brazilian patients, including key entries such as Patient ID, Appointment ID, appointment scheduling date, appointment date, and demographic information including gender, age, and neighborhood. Furthermore, it encompasses data regarding the SMS reminder status, denoted as 1/0 for Yes/No, along with clinical history variables (Hypertension, Diabetes, Alcoholism, and Handicap), each represented as 1/0 to indicate their presence or absence. The dataset includes scholarship status, indicated as 1/0 for Yes/No, and the no-show status, which is the goal variable, denoted as Yes/No. This dataset accurately represents the realities of Brazilian public healthcare, rendering it an optimal resource for analyzing appointment adherence trends.

5. Results and Discussion

5.1 Numerical Results

The Voting ensemble, integrating CatBoostClassifier, RandomForestClassifier, and LogisticRegression through a soft-voting methodology, produced decent predictive performance. The accuracy was 46.54%, while the sensitivity was 73.25%. Nevertheless, specificity was low at 39.79%, and the F1-score was at 0.36, indicating challenges in accurately classifying those who attended their visits. The ROC AUC was 0.61, signifying little discriminative ability.

The Bagging ensemble, with a gradient-boosting base estimator, attained an accuracy of 52.36%, reflecting a modest enhancement. Sensitivity was 65.93%, while specificity was 48.92%. The ROC AUC stabilized at approximately 0.60, while the F1-score registered at 0.36, indicating that although the technique somewhat enhanced balanced classification, the overall discriminative capacity remained constrained.

The Stacking ensemble, which integrates base learners with a logistic regression meta-learner, had its peak performance in recall, with sensitivity reaching 76.79%. This indicates that the model was most proficient at identifying no-shows among the preceding three models. Nonetheless, its specificity decreased to 33.54%, and its

ROC AUC was 0.59. The results indicate a trade-off: the model identified several no-shows but also generated a significant number of false positives by mistakenly predicting no-shows.

In contrast, the Stacked LightGBM ensemble significantly outperformed the earlier approaches in discriminative power: the model reached a ROC AUC of 0.70, with accuracy at 52.59%. Most notably, its sensitivity was 93.44%, indicating an exceptional ability to predict no-shows. While its specificity of 42.25% was lower compared to other models, its F1-score of 0.44 suggested that it struck a balance between identifying no-shows and reducing false positives.

The AutoGluon Tabular pipeline achieved the highest ROC AUC at 0.76 and an accuracy of 80.84%, demonstrating strong overall discrimination. However, this came at the cost of sensitivity: only 15.23% of actual no-shows were identified. The model produced a specificity of 97.44%, indicating it very reliably classified patients who attended. The F1-score was 0.24, and the precision-recall AUC was 0.45, indicating inadequate recall for the minority class despite high confidence in the majority class predictions. This trend is probably attributable to the class-weighting or sample-weighting method employed by AutoGluon (inverse-frequency weighting), which skews the model towards majority-class predictions.

The results demonstrate a trade-off between sensitivity and specificity; wherein prioritizing recollection may result in an elevation of false positives (incorrectly predicting no-shows when the patient actually attends). This trade-off is characteristic of imbalanced classification jobs and is a crucial factor in healthcare applications (Table 3).

Table 3. Performance metrics of five different machine learning models

Model	Accuracy	Sensitivity (Recall)	Specificity	F1 score	ROC AUC	PR AUC
Catboost Voting ensemble	0.465439247	0.732526882	0.397857386	0.356267364	0.608539416	0.277739589
Catboost Bagging ensemble	0.523568262	0.659274194	0.489230246	0.358508954	0.604276668	0.27158353
Catboost Stacking ensemble	0.422735909	0.767921147	0.335392813	0.349492787	0.585592434	0.259192482
Stacked LightGBM ensemble	0.525875328	0.934363799	0.422514454	0.443181214	0.701774622	0.59400896
AutoGluon	0.808423053	0.152329749	0.974436005	0.243074173	0.763069012	0.446834782

5.2 Graphical Results

Comparing ROC curves across the pipelines (Fig 1-5), the Voting, Bagging, and Stacking ensembles had AUCs of 0.60, 0.59, and 0.61 respectively; the Stacked LightGBM reached 0.70; and AutoGluon achieved 0.76. The gradual

improvement in AUC reflects the increasing sophistication of the methods from simple ensemble voting/bagging to meta-learning and automatic hyperparameter optimization (Figure 1- Figure 6).

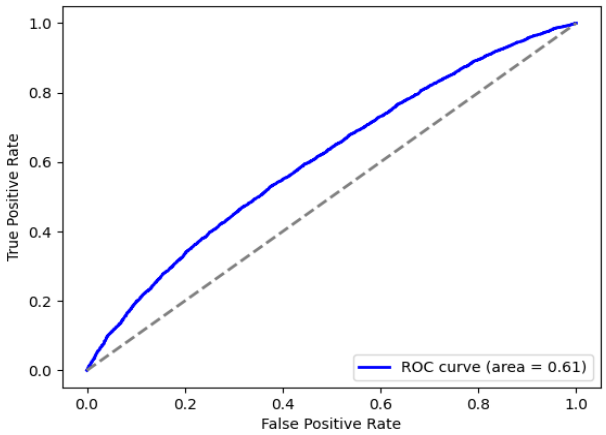

Figure 1. Receiver Operating Characteristic (ROC) curve for Voting classification model

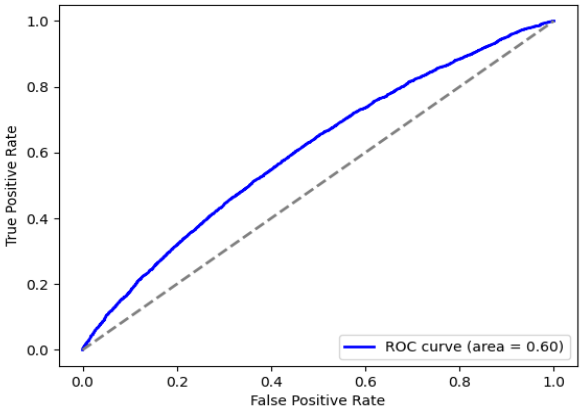

Figure 2. Receiver Operating Characteristic (ROC) curve for Bagging classification model

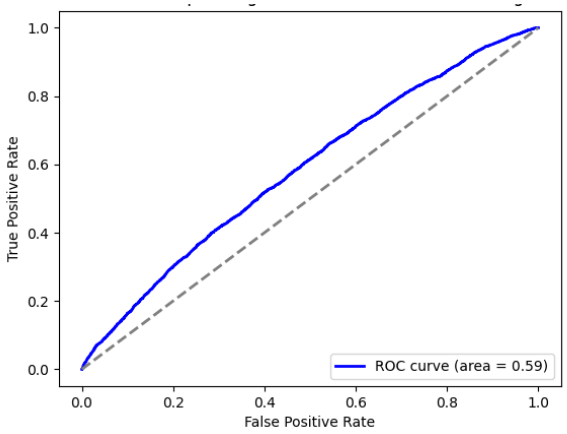

Figure 3. Receiver Operating Characteristic (ROC) curve for Stacking classification model

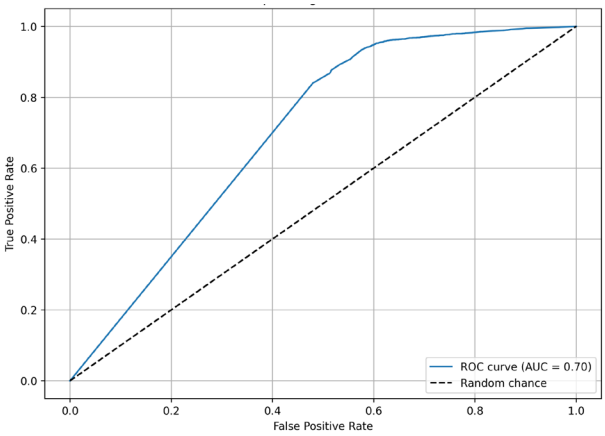

Figure 4. Receiver Operating Characteristic (ROC) curve for Stacked LightGBM model

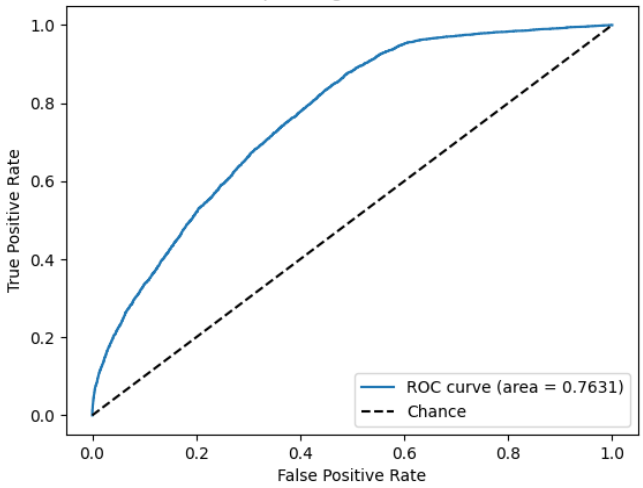

Figure 5. Receiver Operating Characteristic (ROC) curve for AutoGluon model

SHAP analysis performed for the AutoGluon model revealed that among all features, Age, whether an SMS was received, and scheduling features such as the original scheduling day had the strongest influence on the model's output, depicted on Fig 6. The SHAP summary plot distinctly illustrated how these factors influenced the estimated chance of no-show, providing clear insight into the determinants of the model's judgments. SHAP values indicates the feature importance, using equation 6.

$$\phi_i = \sum_{S \subseteq N\{i\}} \frac{|S|!\ (|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\}) - v(S)] \tag{6}$$

Where ϕ_i is the SHAP value of feature i, N is the set of all features, S is a subset of features excluding i, $v(S)$ is the value function for the subset S, and $|S|$ and $(|N| - |S| - 1)!$ are factorials representing permutations of feature subsets.

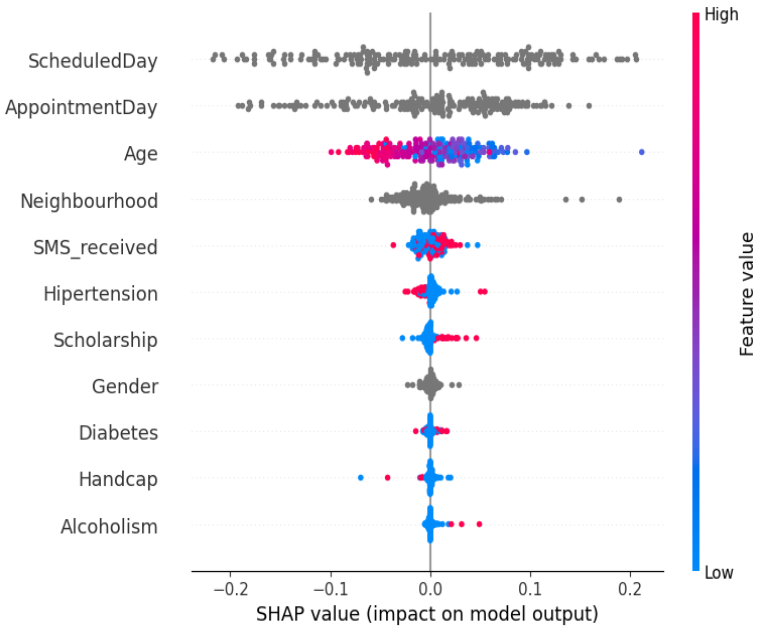

Figure 6. SHAP summary plot with respect to the feature value

5.3 Proposed Improvements

While our work provides a robust basis for predicting no-shows in a Brazilian outpatient context, numerous opportunities exist for further enhancement and extension of this research. Future research may improve forecast accuracy by incorporating further contextual factors such as weather conditions, local transit accessibility, and socioeconomic indicators like income, employment status, housing conditions, and neighborhood-level deprivation indices. Integrating these contextual factors can enhance the alignment of predictive models with actual obstacles to appointment attendance, as evidenced by similar techniques documented in previous studies (Leiva-Araos et al., 2025). Furthermore, transcending the presumption that each appointment is an isolated occurrence, further research might analyze temporal dynamics and longitudinal patient behavior by integrating previous attendance records, the duration since the last visit, and the frequency of appointments. These methodologies, encompassing time-series analysis, recurrent neural networks, and survival analysis, are adept at capturing the dynamic risk of no-shows across time and may produce more precise and individualized predictions. Furthermore, future research could transcend conventional classification metrics by implementing cost-sensitive and optimization-focused modeling frameworks that explicitly assess the economic and operational ramifications of false positives, such as excessive overbooking, and false negatives, such as unutilized appointment slots. This approach would facilitate the optimization of scheduling strategies, including overbooking policies and dynamic reminder distribution (Leiva-Araos et al., 2025). Finally, it would be important to use datasets from different geographic areas, healthcare settings (public vs. private), and patient groups to test the model's robustness and generalizability. Cross-site validation would make the case for wider use much stronger.

6. Conclusion

This study evaluated three machine learning techniques for predicting patient appointment no-shows in Brazilian healthcare. The AutoGluon model had the highest accuracy and ROC AUC, establishing it as the most feasible choice for practical use. However, challenges remain, particularly in balancing sensitivity and specificity, which are crucial in healthcare settings, since overbooking can lead to resource exhaustion, whereas underbooking may jeopardize patient care. The findings demonstrate that machine learning models, particularly those employing automated frameworks like AutoGluon, may significantly optimize appointment scheduling and resource allocation, hence enhancing operational efficiency and continuity of patient care. Future research should focus on improving model generalizability, incorporating more contextual information, and enabling the actual application of predictive models in healthcare practices.

References

Abushaaban, E. and Agaoglu, M., Medical appointment no-show prediction using machine learning techniques, in *Proc. 2nd Int. Conf. on Computing and Machine Intelligence (ICMI)*, 2022. DOI: 10.1109/ICMI55296.2022.9873652.

Alaidah, A., Alamoudi, E., Shalabi, D., AlQahtani, M., Alnamshan, H. and Abubacker, N. F., Mining and predicting no-show medical appointments using hybrid sampling technique, *Lecture Notes in Networks and Systems*, vol. 204, pp. 315–333, 2021. DOI: 10.1007/978-981-16-1089-9_27.

Almeida, R., Silva, N. A. and Vasconcelos, A., A machine learning approach for real-time prediction of last-minute medical appointment no-shows, in *Proc. 14th Int. Conf. on Health Informatics (HEALTHINF)*, pp. 328–336, 2021. DOI: 10.5220/0010221903280336.

Alshammari, A., Almalki, R. and Alshammari, R., Developing a predictive model of predicting appointment no-show using machine learning algorithms, *Journal of Advances in Information Technology*, vol. 12, no. 3, pp. 234–239, 2021. DOI: 10.12720/jait.12.3.234-239.

Alshammari, A., Alotaibi, F. and Alnafrani, S., Prediction of outpatient no-show appointments using machine learning algorithms for pediatric patients in Saudi Arabia, *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, 2024.

Alshammari, R., Daghistani, T. and Alshammari, A., The prediction of outpatient no-show visits using deep neural networks from large data, *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 10, 2020.

Amalina, N. N., Boateng Ofori-Amanfo, K. and An, H., A multi-head attention soft random forest for interpretable patient no-show prediction, n.d.

Barrera Ferro, D., Brailsford, S., Bravo, C. and Smith, H., Improving healthcare access management by predicting patient no-show behaviour, *Decision Support Systems*, vol. 138, 2020. DOI: 10.1016/j.dss.2020.113398.

Batool, T., Abuelnoor, M., El Boutari, O., Aloul, F. and Sagahyroon, A., Predicting hospital no-shows using machine learning, in *Proc. IEEE Int. Conf. on Internet of Things and Intelligence Systems (IoTaIS)*, pp. 142–148, 2021. DOI: 10.1109/IoTaIS50849.2021.9359692.

Can machine learning predict patient no-shows?, *AAPC Knowledge Center*, Retrieved Dec. 15, 2025.

Daghistani, T., AlGhamdi, H., Alshammari, R. and AlHazme, R. H., Predictors of outpatients' no-show: big data analytics using Apache Spark, *Journal of Big Data*, vol. 7, no. 1, 2020. DOI: 10.1186/s40537-020-00384-9.

Deina, C., Fogliatto, F. S., da Silveira, G. J. C. and Anzanello, M. J., Decision analysis framework for predicting no-shows to appointments using machine learning algorithms, *BMC Health Services Research*, vol. 24, no. 1, 2024. DOI: 10.1186/s12913-023-10418-6.

Dunstan, J., Villena, F., Hoyos, J. P., Riquelme, V., Royer, M., Ramírez, H. and Peypouquet, J., Predicting no-show appointments in a pediatric hospital in Chile using machine learning, *Health Care Management Science*, vol. 26, no. 2, pp. 313–329, 2023. DOI: 10.1007/s10729-022-09626-z.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M. and Smola, A., AutoGluon-Tabular: robust and accurate AutoML for structured data, 2020. arXiv:2003.06505.

Fan, G., Deng, Z., Ye, Q. and Wang, B., Machine learning-based prediction models for patients' no-show in online outpatient appointments, *Data Science and Management*, vol. 2, pp. 45–52, 2021. DOI: 10.1016/j.dsm.2021.06.002.

How much each year do no-shows cost the U.S. healthcare system?, Retrieved Dec. 15, 2025.

Incze, E., Holborn, P., Higgs, G. and Ware, A., Using machine learning tools to investigate factors associated with trends in no-shows in outpatient appointments, *Health and Place*, vol. 67, 2021. DOI: 10.1016/j.healthplace.2020.102496.

Kaplan-Lewis, E. and Percac-Lima, S., No-show to primary care appointments: why patients do not come, *Journal of Primary Care and Community Health*, vol. 4, no. 4, pp. 251–255, 2013. DOI: 10.1177/2150131913498513.

Leiva-Araos, A., Contreras, C., Kaushal, H. and Prodanoff, Z., Predictive optimization of patient no-show management in primary healthcare using machine learning, *Journal of Medical Systems*, vol. 49, no. 1, 2025. DOI: 10.1007/s10916-025-02143-w.

Liu, D., Shin, W. Y., Sprecher, E., Conroy, K., Santiago, O., Wachtel, G. and Santillana, M., Machine learning approaches to predicting no-shows in pediatric medical appointments, *npj Digital Medicine*, vol. 5, no. 1, 2022. DOI: 10.1038/s41746-022-00594-w.

Marbouh, D., Khaleel, I., Shanqiti, K. Al, Tamimi, M. Al, Simsekler, M. C. E., Ellahham, S., Alibazoglu, D. and Alibazoglu, H., Evaluating the impact of patient no-shows on service quality, *Risk Management and Healthcare Policy*, vol. 13, pp. 509, 2020. DOI: 10.2147/RMHP.S232114.

Medical appointment no-shows dataset, *Kaggle*, Retrieved Dec. 15, 2025.

Nguyen, P. T., Dang, D. T. and Nguyen, V. D., A robust deep learning technique for no-show prediction in hospital appointments, *Lecture Notes on Data Engineering and Communications Technologies*, vol. 184, pp. 3–18, 2023. DOI: 10.1007/978-3-031-43247-7_1.

Osorio, F. O., Gomez, S. P., Sanchez, D. E. R., Fernandez, R. R., Tabares-Soto, R., Bravo-Ortíz, M. A. and Suarez, G. A. C., Predicting no-shows at outpatient appointments in internal medicine using machine learning models, *PeerJ Computer Science*, vol. 11, pp. 1–29, 2025. DOI: 10.7717/peerj-cs.2762.

Pereira, J. S. B., Valdrighi, G. and Raimundo, M. M., M²FGB: A min–max gradient boosting framework for subgroup fairness, 2025. arXiv:2504.12458.

Salazar, L. H. A., Leithardt, V. R. Q., Parreira, W. D., da Rocha Fernandes, A. M., Barbosa, J. L. V. and Correia, S. D., Application of machine learning techniques to predict a patient's no-show in the healthcare sector, *Future Internet*, vol. 14, no. 1, 2022. DOI: 10.3390/fi14010003.

Seghier, M. L., Ten simple rules for reporting machine learning methods implementation and evaluation on biomedical data, *International Journal of Imaging Systems and Technology*, vol. 32, no. 1, pp. 5–11, 2022. DOI: 10.1002/ima.22674.

Srinivas, S. and Salah, H., Consultation length and no-show prediction for improving appointment scheduling efficiency at a cardiology clinic, *International Journal of Medical Informatics*, vol. 145, 2021. DOI: 10.1016/j.ijmedinf.2020.104290.

Valero-Bover, D., González, P., Carot-Sans, G., Cano, I., Saura, P., Otermin, P., Garcia, C., Gálvez, M., Lupiáñez-Villanueva, F. and Piera-Jiménez, J., Reducing non-attendance in outpatient appointments: predictive model development, validation, and clinical assessment, *BMC Health Services Research*, vol. 22, no. 1, 2022. DOI: 10.1186/s12913-022-07865-y.

Wakayama, T. and Sugasawa, S., Ensemble prediction via covariate-dependent stacking, 2025. arXiv:2408.09755.

Zhang, Y. and Chen, Y., Reducing no-show rates through digitalization: AI and SMS-based interventions for better patient adherence, in *Proc. Int. Conf. on Digital Economy and Intelligent Computing (DEIC)*, pp. 123–127, 2025. DOI: 10.1145/3746972.3746993.

Biographies

Md. Ashhab Rahman is an undergraduate student in Industrial and Production Engineering at the Military Institute of Science and Technology (MIST). His research interests include machine learning, operations research and supply-chain management.

Tasnimul Hasan Nishorgo is an undergraduate student in Industrial and Production Engineering at the Military Institute of Science and Technology (MIST). His research interests include machine learning, supply chain optimization and sustainable manufacturing.

Tanjeel Ahmed Bin Zaman is a lecturer in the Department of Industrial and Production Engineering (IPE) at the Military Institute of Science and Technology (MIST). His research interests encompass areas such as production optimization, supply chain management, lean manufacturing, and operations research. With a strong focus on enhancing operational efficiency, he aims to contribute to both academia and industry through his work.