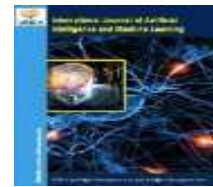




International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: <https://www.svedbergopen.com/>



Research Paper

Open Access

Improved Prolonged Length of ICU Stay Prediction using Stacked Machine Learning Models

Arun Kumar Gautam¹, Ankit Chauhan², Shikha Verma³

¹Assistant Professor, Department of Computer Science, Ram Lal Anand College, University of Delhi, Delhi, India

²Student, Department of Computer Science, Ram Lal Anand College, University of Delhi, Delhi, India

³Assistant Professor, Department of Computer Science, Ram Lal Anand College, University of Delhi, Delhi, India

Abstract

The most important component of the modern healthcare industry and the hospital are Intensive Care Units (ICUs). The optimal management of the resources with respect to the prolonged length of stay (pLOS) of patients in the ICUs is equally important. The objective of this study was to develop improved prediction models using stacked double layered machine learning models for the prolonged length of stay (pLOS) of ICU patients. The analysis of the eICU-CRD dataset used in the study suggests that Gradient Boosted Decision Tree (GBDT), RF (Random Forest), and DL (Deep Learning) demonstrate great potential and are recommended machine learning models for the intended application. The stacked layers approach utilizes a total combination of nine stacked models using GBDT, RF and DL. These models were compared based on area under the receiver operating characteristic curve (AUROC), area under the precision-recall curve (AUPRC), Brier score and calibration curve. Out of all the proposed models, the GBDT-GBDT layered model (AUPRC = 0.783, AUROC = 0.595, Brier score = 0.140) achieved the best performance. According to the calibration curve, RF-GBDT and DL-GBDT model performance was better among the nine models. All the proposed stacked models performed better than all the base layer models (GBDT, RF and DL), which implies the significance of using multi-layered models rather than single layer models for better predictions.

Keywords

Stacked layered approach; Gradient-Boosted Decision Trees (GBDT); RF (Random Forest); DL (Deep Learning); Model Calibration; Clinical Decision Support

1. Introduction

Biomedical science, an expanding discipline, comprises the delivery of healthcare, advancement in technology and scientific research. Biomedical sciences derive biomedical informatics, which is the information science as implemented or studied in the biomedicine context [1][69]. Combining biological knowledge with state-of-the-art technology has led to notable advances in clinical results and mortality rates [2]. Research in this field has led to the development of life-saving drugs, innovative treatment strategies, and precise healthcare procedures. As diseases grow more complex, there is a rising need for the use of data-driven methods like machine learning (ML), artificial intelligence (AI), and deep learning (DL) to increase the efficacy and accuracy of medical decisions. Utilising these computational methods has become crucial as the healthcare industry develops, in order to handle gigantic volumes of medical data [3].

One of the most crucial areas of modern healthcare is the Intensive Care Unit (ICU), where patients with serious health conditions receive specialized medical treatment. ICUs play a very important role in treating organ failures, serious infections, and post-surgical recoveries. Given the high risk of ICU patients, timely and precise decision-making is crucial to ensure that they receive the best treatment possible. The digitization of ICUs has resulted in the generation of enormous volumes of real-time data every day [4]. This collected data is useful for developing focused treatment plans, finding trends in diseases, and enhancing healthcare facilities [5]. Storing and organizing this data effectively is essential, but it can be extremely difficult to analyse manually [6] and thus requires advanced analysis techniques like data science to enhance medical decision-making and patient outcomes [7].

The storage and management of this enormous patient data in healthcare institutions has been completely revolutionized by Electronic Health Records (EHRs). EHRs are digitally stored versions of patient medical records which allow the smooth exchange of information between healthcare providers by offering a structured digital representation of patient records [8]. EHRs are a valuable resource for medical research as electronically stored records are considerably easier to analyse and enable the identification of trends in diseases as well as medication efficacy. Despite the availability of large-scale electronic healthcare data, there are several challenges associated with its effective use. It is critical to recognise that large amounts of data may introduce serious biases and possibly reduce the credibility of a study. The most common cause of such bias is missing data, but selection, information, and ascertainment errors can significantly affect available data and outcomes too [9]. Errors and biases may occur during the study's design, data collection, analysis, or conclusion phases [9, 10].

Addressing these difficulties necessitates the use of sophisticated analytical approaches like statistical techniques and natural language processing (NLP), which are capable of efficiently processing massive, complex, and diverse datasets. In this context, ML and DL have emerged as effective techniques for analysing and understanding complex healthcare data. ML methods, such as decision trees, support vector machines, and tree ensemble methods, enable predictive modelling and data-driven decision-making. DL, a subfield of machine learning, uses artificial neural networks (ANNs) to uncover hierarchical patterns from vast datasets, making it especially useful for natural language processing, medical imaging, and genomics [11, 12]. The capability of ML and DL models to automatically learn from data and improve over time has made them essential in dealing with the issues of biological data processing. This is the reason that the application of ML and DL in biomedical research and healthcare has resulted in groundbreaking advances [13, 14]. These techniques are used in the healthcare industry to improve patient outcomes and treatment quality. ML applications are used for disease prediction, visualization of biomedical data, personalized treatment options, among other things, while DL applications are used for drug development, imaging in medicine, studying mental health, and many other purposes [15, 16]. The use of these new techniques continues to improve diagnostic accuracy, treatment efficacy, and overall patient care, thereby changing the future of modern medicine.

2. Preliminaries

2.1. Machine Learning Models

Gradient Boosted Decision Tree (GBDT) is an ensemble decision tree classifier model where the decision trees are trained sequentially [17, 18, 19, 20]. GBDT uses the method of boosting in which several weak learning models, decision trees in this case, are combined to form a strong learning model. Every new decision tree tries to minimize the errors or loss function of the previous decision tree using the gradient descent algorithm, which significantly minimizes the errors of the model as a whole. During each iteration, the algorithm uses the current model's predictions to compute the gradient of the model's loss function and then a new model is trained to minimize this loss. After adding this model's predictions to the ensemble, the procedure goes on till the stoppage requirement is satisfied. In comparison to other models, GBDT models are typically more accurate because of this continuous minimisation of loss. The GradientBoostingClassifier from the Python sklearn.ensemble package was used in this study [21].

Random Forest (RF) is also an ensemble decision tree classifier model where the decision trees are trained parallelly [20, 22, 23, 24] unlike the GBDT model. In the RF model, each decision tree is trained on independent but identically distributed and randomly generated samples of the original dataset. Each tree casts a vote to predict the class of the input data and the votes of each tree are combined to get the final output. The class which gets the majority votes is the final prediction. The randomness in selecting samples for each tree helps to prevent the overfitting of the model, making the predictions more accurate and reliable. The RandomForestClassifier from the Python sklearn.ensemble package was used in this study [21].

Deep Learning (DL), which is a subfield of ML, learns from data by using ANNs [25, 26, 27]. ANNs draw inspiration from the way the human brain processes information [28, 29]. In this study we used the MLP Classifier of the sklearn.neural_network package of Python [21]. Multilayer perceptron (MLP) is a category of ANNs which has multiple layers of connected neurons and is renowned for its capability to represent complicated non-linear relationships in the data [30, 31].

2.2. Evaluation Metrics

The Area Under the Receiver Operating Characteristics Curve (AUROC) is a common performance metric which is used to summarize the information contained in the receiver operating characteristics (ROC) curve [32]. The ROC curve is a very useful technique to visualize the performance of supervised classification models [33, 34] and the area under the curve (AUC) helps in summarizing the performance which helps in selecting the best performing model. In this study we used the auc_roc_score of the sklearn.metrics package of Python [21].

The Area Under the Precision-Recall Curve (AUPRC) score is another useful performance metric for analysing the performance of classification models, which is calculated by integrating the precision-recall curve (PRC) [35]. Precision is defined as the measure of the number of true positive predictions out of all positive predictions made by the model, while Recall is defined as the measure of the number of true positive predictions made by the model out of all the possible positive predictions. The PRC visualises the correlation between precision and recall for every threshold value. In this study we used the average_precision_score of the sklearn.metrics package of Python [21].

The Brier Score is a metric which is used to evaluate the probabilistic predictions of a classification model [36]. The Brier Score is used for binary classification models and it is calculated by measuring the mean squared difference between the target and the predicted probability [37]. The value of Brier Score defines how well the predicted probabilities are calibrated. The lower the Brier Score, the better, which is why it is also termed Brier Score Loss. In this study we used the brier_score_loss of the sklearn.metrics package of Python [21].

3. Literature Review

In recent times, the upsurge of machine learning and deep learning models has led to the development of many prediction models in the biomedical context using various approaches like standard regression models, machine

learning models and neural network based models. Some models were developed using data from a single centre while some used data from multiple centres. Although most of the models included in our study used data for general ICU patients, some also used data of patients with specific conditions or undergoing some procedures.

For instance, Herman et al. [38] used data of ICU patients from a single-centre undergoing isolated coronary artery bypass grafting (CABG) to develop a logistic regression (LR) based model for predicting pLOS-ICU with an AUROC score of 0.78. Rotar et al. [42] also developed a least absolute shrinkage and selection operator (LASSO) regression model for predicting pLOS-ICU with an AUROC score of 0.72 using data of ICU patients following CABG from a single centre. Hachesu et al. [51] used data of patients with coronary artery disease (CAD) from a single centre to develop an ICU-LOS classification model which can classify patients into three classes according to their ICU-LOS using decision tree C5.0, support vector machine (SVM) and artificial neural network (ANN) models, with SVM achieving the maximum accuracy of 0.96.

Many models were developed to predict pLOS-ICU, but there are also models which can predict ICU mortality, acute kidney injury (AKI), etc. For example, Qian et al. [39] used data of general ICU patients from a single centre for predicting AKI in patients using models like LR, SVM, RF, extreme gradient boosting (XGBoost), light gradient boosting machine (LightGBM) and convolutional neural network (CNN), where the LightGBM model outperformed all other models with an AUROC of 0.905. Lin et al. [44], Meiring et al. [45] and Viton et al. [46] all developed models for predicting ICU mortality, where only Meiring et al. [45] used a multi-centre dataset while others used single-centre datasets only. Lin et al. [44] used data of patients with AKI from a single centre to develop four ICU mortality prediction models using RF, SVM, ANN and Customized SAPS II, with a maximum AUROC score of 0.866 achieved using RF. Meiring et al. [45] developed LR, adaptive boosting (AdaBoost), RF, SVM, a single layer averaged neural network (avNNet), DL using Keras, and APACHE II models with a maximum AUROC of 0.883 achieved by RF. Viton et al. [46] only developed a DL model based on CNN with an AUROC of 0.85.

Chen et al. [47] used data of general ICU patients from a single centre to develop models like MLP, SVM, k-nearest neighbours (KNN), RF, XGBoost, LightGBM, transformer based CNN (TCNN), two-dimensional CNN (2D-CNN) and one-dimensional multi-scale network (1D-MSNet) for predicting LOS-ICU, where 1D-MSNet outperformed every other model with an R^2 value of 0.57. Rocheteau et al. [49] also developed a Temporal Pointwise CNN model for predicting LOS-ICU but using multi-centre data, achieving an R^2 value of 0.40.

For developing models for pLOS-ICU prediction, almost all the models included in our study used data of general ICU patients from a single centre. For example, Zoller et al. [40] developed models using indocyanine green plasma disappearance rate (ICG-PDR), mid-regional pro-atrial natriuretic peptide (MR-proANP), proadrenomedullin (pro-ADM), and copeptin, as well as SAPS II, for predicting pLOS-ICU. All these models were outperformed by the SAPS II model except the ICG-PDR model, with an AUROC of 0.73.

Iwase et al. [43] developed models for predicting ICU mortality, and short and long ICU stay using data of general ICU patients from a single centre. For short and long ICU stay prediction, the models developed were RF and LR using APACHE II or SOFA score, where the RF model had better prediction for both short and long ICU stay with an AUROC of 0.881 (Short ICU stay) and AUROC of 0.889 (Long ICU stay). For ICU mortality prediction, the models developed were RF, XGBoost, NN and LR using APACHE II or SOFA score, where the RF model again outperformed all other models with an AUROC of 0.945.

Houthoofd et al. [41] developed models like ANN, SVM, KNN, RF, AdaBoost and decision tree based classification and regression tree (CART) using data of general ICU patients from a single centre to predict pLOS-ICU. Out of all these models, the SVM model performed the best with an AUROC of 0.82. Navaz et al. [48] used data of general ICU patients from a single centre to develop a pLOS-ICU prediction model using decision tree, but the model did not perform well, with only 59% accuracy.

Ma et al. [50] also used data of general ICU patients from a single centre but developed a model for predicting whether patients can be discharged from the ICU within 10 days. The model was developed by combining just-in-time learning (JITL) and one-class extreme learning machine (one-class ELM), referred to in short as one-class JITL-ELM, where the model had an AUROC of 0.851.

Wu et al. [54] was the only study which used a multi-centre dataset to predict pLOS-ICU for general ICU patients. They developed models like GBDT, SVM, RF, DL using MLP and Customized SAPS II for the task, where the GBDT model outperformed all other models with an AUROC of 0.742 for internal validation. They also validated their models externally using a single-centre dataset independent from their original dataset used for training and internal validation of models. The GBDT model still outperformed all other models, with an AUROC value of 0.747 for external validation. Table 1 summarizes the basic characteristics of some studies included in our research. Although many of the studies included in our research for predicting pLOS-ICU developed models with very good performance (maximum AUROC of 0.851 [50]), all of these models were developed using data only from a single centre except for one [54], which used multi-centre data for pLOS-ICU model development and also used an independent single-centre dataset for external validation but achieved an AUROC score of just 0.747. Multi-centre data tends to be more heterogeneous than their single-centre counterparts, and single-centre datasets are more affected by centre-specific biases. Thus, even when single-centre data seems to be a better fit for training models, some studies conclude that models fitted on multi-centre data perform better than models fitted on single-centre data when the prediction error is the concern. But single-centre

data show better performance when the chance of successful validation and false discovery rate is the concern [52]. While multi-centre data has advantages like more generalized data and larger sample sizes, it also has disadvantages, such as different procedures being used by different centres, which results in single-centre data being more reliable in some cases [53].

Table 1. Summary of studies using machine learning models for ICU-related predictions.

Dataset Population	Dataset Type	Sample Size	Study	Target	Models	Evaluation Metrics
ICU Patients undergoing isolated CABG	Single-centre	3,483	Herman et al. [38]	pLOS-ICU	LR	AUROC: 0.78
General ICU Patients	Single-centre	17,205	Qian et al. [39]	AKI	LR, SVM, RF, XGBoost, LightGBM, CNN	AUROC: 0.90 (LightGBM)
General ICU Patients	Single-centre	110	Zoller et al. [40]	pLOS-ICU	ICG-PDR, SAPS II, MR-proANP, pro-ADM, Copeptin	AUROC: 0.73 (ICG-PDR)
General ICU Patients	Single-centre	14,480	Houthoofd et al. [41]	pLOS-ICU	ANN, KNN, SVM, CART, RF, AdaBoost	AUROC: 0.82 (SVM)
ICU Patients following CABG	Single-centre	3,283	Rotar et al. [42]	pLOS-ICU	LASSO	AUROC: 0.72
General ICU Patients	Single-centre	12,747	Iwase et al. [43]	ICU Mortality	RF, XGBoost, NN, LR using APACHE II or SOFA	AUROC: 0.95 (RF)
Patients with AKI	Single-centre	19,044	Lin et al. [44]	ICU Mortality	RF, SVM, ANN, Customized SAPS II	AUROC: 0.87 (RF)
General ICU Patients	Multi-centre	21,911	Meiring et al. [45]	ICU Mortality	LR, AdaBoost, RF, SVM, avNNet, DL, APACHE II	AUROC: 0.88 (DL)
General ICU Patients	Single-centre	13,000+	Viton et al. [46]	ICU Mortality	DL	AUROC: 0.85
General ICU Patients	Single-centre	40,000	Chen et al. [47]	LOS-ICU	MLP, SVM, KNN, RF, XGBoost, LightGBM, TCNN, 2D-CNN, 1D-MSNet	R ² : 0.57 (1D-MSNet)
General ICU Patients	Single-centre	40,426	Navaz et al. [48]	pLOS-ICU	Decision Tree	Accuracy: 0.59
General ICU Patients	Multi-centre	168,577	Rocheteau et al. [49]	LOS-ICU	Temporal Pointwise CNN	R ² : 0.40
General ICU Patients	Single-centre	4,000	Ma et al. [50]	pLOS-ICU	One-Class JITL-ELM	AUROC: 0.85
Patients with CAD	Single-centre	4,948	Hachesu et al. [51]	LOS-ICU Classification	Decision Tree C5.0, SVM, ANN	Accuracy: 0.96 (SVM)
General ICU Patients	Multi-centre	160,238	Wu et al. [54]	pLOS-ICU	Customized SAPS II, GBDT, RF, SVM, DL	AUROC: 0.75 (GBDT)

4. Proposed Work

4.1. Dataset Description

The dataset used for the development of the prediction models was the eICU Collaborative Research Database (eICU-CRD) [55]. The eICU-CRD is a large publicly available multi-centre critical care database which is maintained by the Laboratory for Computational Physiology (LCP) at the Massachusetts Institute of Technology (MIT) and the eICU Research Institute (eRI). The patients included in the dataset were selected by stratified random sampling. The database

comprises 200,859 unit admissions for 139,367 different patients admitted at one of 208 different hospitals between the years 2014 to 2015 across the US. The database includes vital signs, laboratory measurements, medications, admission diagnosis, treatment information, patient history, care plan information, etc. The data are organised into different tables corresponding to the type of data contained in each table. All tables in the database are de-identified to meet the provisions of the US Health Insurance Portability and Accountability Act (HIPAA) [56], protecting the privacy of both the patients and the institutions which contributed to the data.

4.2. Framework

The models proposed in this study use the same features as used in the Simplified Acute Physiology Score (SAPS II), which is used for mortality prediction of patients. The preprocessing steps for feature extraction and data cleaning are summarised in Figure 1. The database was used to extract a total of 17 features which consist of heart rate, age, body temperature, systolic blood pressure, Glasgow Coma Score (GCS), PaO₂/FiO₂ ratio, urine output, bilirubin level, Blood Urea Nitrogen (BUN) level, potassium level, sodium level, bicarbonate level, White Blood Cells (WBC), three underlying diseases namely acquired immunodeficiency syndrome (AIDS), haematological malignancy, and metastatic cancer, and the type of admission (scheduled surgical, unscheduled surgical, or medical) [57]. For the time-stamped features — systolic blood pressure, potassium level, and bicarbonate level — the extreme values (both minimum and maximum) within the first 24 hours of admission were selected for a patient. For the remaining time-stamped features, the values given in the APACHE tables were used. The tables used for extracting these features are apachePredVar, apacheApsVar, vitalAperiodic, and lab. Apart from these, the target variable, ICU-LOS, representing the actual ICU length of stay extracted from the apachePatientResult table, was also used for the training and evaluation phases of the models.

After extracting the features from the database, they were combined into a single table. The data was then pre-processed for use in developing the prediction models. The preprocessing involved extracting patient records aged from 18 to 90 years. Records were excluded if the following criteria were satisfied: the patient died within the ICU, as patients who die within the ICU might have differing patterns than those who survived [58]; the ICU-LOS variable is missing for a record; or the feature missing rate for a record is more than 30%. The remaining missing values for each feature were filled with the median values of the corresponding feature. The elective surgery column was used to identify the type of admission of patients as follows: if the value of elective surgery was 1, then scheduled surgical; if the value was 0, then unscheduled surgical; and medical if the value was missing, which was filled with the value 2 to represent medical admission type. Finally, we transformed the ICU-LOS variable into a categorical type variable where the value was changed to 1 if the value of ICU-LOS was greater than 3 days and 0 otherwise, as prolonged ICU length of stay is defined as length of ICU stay more than the mean length of stay, which is 3 days in the United States [54].

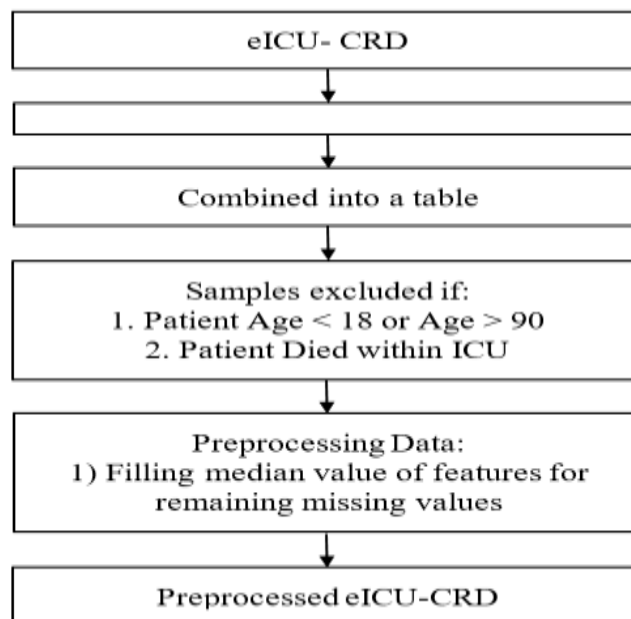


Fig. 1. Preprocessing of eICU Dataset

4.3. Mathematical Formulation of the Proposed Framework

This section presents a formal mathematical description of the two-layer stacked framework introduced in Section 4.2, prior to detailing the implementation of each stacked configuration in Sections 4.4 to 4.7.

4.3.1. Problem Definition

Let the pre-processed dataset be denoted
 $D = \{(x_i, y_i)\}_{i=1}^N$, $x_i \in \mathbb{R}^{17}$, $y_i \in \{0, 1\}$

where $N = 115,756$, x_i is the 17-dimensional feature vector for patient i , and $y_i = 1$ if ICU-LOS > 3 days (pLOS-ICU) and 0 otherwise. The dataset D is partitioned into a training set D_{train} (70%, $N_{\text{tr}} = 81,029$) and a test set D_{test} (30%, $N_{\text{te}} = 34,727$).

4.3.2. Layer 1: Base Models

Each base model learns a mapping $f_b(1) : \mathbb{R}^{17} \rightarrow [0, 1]$, $b \in \{\text{GBDT}, \text{RF}, \text{DL}\}$, that estimates the posterior probability

$$\hat{p}_b(x) = P(y = 1 | x)$$

(i) Gradient Boosted Decision Tree (GBDT). GBDT builds an additive ensemble of M regression trees h_m fitted sequentially on the negative gradient of a loss function L (binomial deviance):

$$F_0(x) = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{tr}}} L(y_i, \gamma)$$

$$r_{im} = - [\partial L(y_i, F(x_i)) / \partial F(x_i)]_{F=F_{m-1}}$$

$$h_m = \underset{h}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{tr}}} (r_{im} - h(x_i))^2$$

$$\gamma_m = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{tr}}} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i))$$

$$F_m(x) = F_{m-1}(x) + v \cdot \gamma_m \cdot h_m(x)$$

for $m = 1, \dots, M$, where $v \in (0, 1]$ is the learning rate. The final base-layer probability is obtained via the logistic (sigmoid) link:

$$\hat{p}_{\text{GBDT}}(x) = \sigma(F_M(x)) = 1 / (1 + e^{-F_M(x)})$$

(ii) Random Forest (RF). RF trains B decision trees T_b independently and in parallel, each on a bootstrap sample of D_{train} with random feature sub-sampling at every split, and aggregates the class-probability outputs by averaging:

$$\hat{p}_{\text{RF}}(x) = (1 / B) \sum_{b=1}^B T_b(x)$$

(iii) Deep Learning (DL / MLP). For an L -layer multilayer perceptron with weight matrices $W^{(l)}$, biases $b^{(l)}$, and non-linear activation ϕ (ReLU for hidden layers):

$$h^{(1)} = \phi(W^{(1)}x + b^{(1)})$$

$$h^{(l)} = \phi(W^{(l)}h^{(l-1)} + b^{(l)}), \quad l = 2, \dots, L - 1$$

$$\hat{p}_{\text{DL}}(x) = \sigma(W^{(L)}h^{(L-1)} + b^{(L)})$$

trained by minimizing binary cross-entropy loss via backpropagation:

$$L_{\text{DL}} = -(1/N_{\text{tr}}) \sum_{i=1}^{N_{\text{tr}}} [y_i \log \hat{p}_{\text{DL}}(x_i) + (1 - y_i) \log(1 - \hat{p}_{\text{DL}}(x_i))]$$

4.3.3. Layer 2: Stacked Meta-Models

For each base model b , its predicted probability is appended to the original feature vector to form an augmented meta-feature vector:

$$z_i^{(b)} = [x_i; \hat{p}_b(x_i)] \in \mathbb{R}^{18}$$

giving an augmented dataset $D^{(b)} = \{(z_i^{(b)}, y_i)\}$, which is randomly re-partitioned into new 70/30 training and test splits (Figures 2–4). A second-stage (Layer 2) model g_s , $s \in \{\text{GBDT}, \text{RF}, \text{DL}\}$, is then trained on $D^{(b)}_{\text{train}}$ to produce the final stacked prediction:

$$\hat{p}_{b-s}(x) = g_s(z^{(b)}) = g_s([x; \hat{p}_b(x)])$$

This composition yields the nine stacked models developed in this study:

$$M = \{ f_b^{(1)} \circ g_s \mid b, s \in \{\text{GBDT}, \text{RF}, \text{DL}\} \}$$

i.e. GBDT-GBDT, GBDT-RF, GBDT-DL, RF-GBDT, RF-RF, RF-DL, DL-GBDT, DL-RF, and DL-DL, where the composition operator $\circ$ denotes that g_s consumes both the raw feature vector x and the base model's output $\hat{p}_b(x)$ as its input, as illustrated in Figures 2, 3 and 4 for the GBDT, RF and DL base layers respectively.

4.3.4. Evaluation Metrics — Formal Definitions

For predicted probabilities $\hat{p}_i$ against true labels y_i over a test set of size n , the three evaluation metrics used in this study (Section 2.2) are formally defined as:

$$\text{Brier Score} = (1/n) \sum_{i=1}^n (\hat{p}_i - y_i)^2$$

$$\text{AUROC} = \int_0^1 \text{TPR}(t) d(\text{FPR}(t)), \quad \text{AUPRC} = \int_0^1 \text{Precision}(r) dr$$

where t is the classification threshold, $\text{TPR}(t) = \text{TP}(t) / (\text{TP}(t) + \text{FN}(t))$, $\text{FPR}(t) = \text{FP}(t) / (\text{FP}(t) + \text{TN}(t))$, and r denotes recall.

4.3.5. Note on Meta-Feature Construction

It is important to note that the validity of the composition in Eq. (12) rests on the assumption that, for every training instance used to fit the Layer-2 model g_s , the base-model prediction $\hat{p}_b(x_i)$ is generated by a model that did not observe x_i during its own training (e.g. via out-of-fold or held-out prediction). Where this out-of-fold condition is not strictly met in the implementation described in Sections 4.4–4.7, the reported gains of the stacked models over their respective base models should be interpreted with this caveat in mind, and we recommend it as a priority item for methodological refinement in future iterations of this framework (see Section 7).

4.4. Model Development

4.4.1. Base Models

The dataset was split into a training set of size 70% and a test set of size 30%. The training set was used to train the three models, GBDT, RF, and DL, which were used as the base layer models in the stacked model approach. The model parameters were tuned using hyperparameter tuning to get the best fitted models. The fitted models were used to get predictions for the test dataset. The models' predictions are used to evaluate the models against the original target values. The test data features (X_{test}) combined with the fitted base models' predictions were further combined with

the training data to get three new datasets, which were then randomly shuffled and saved in separate files to be used as training and test data for the Layer 2 models.

4.4.2. Stacked Models with GBDT Base

Figure 2 illustrates the process of developing the three stacked models with the GBDT model as base. The saved dataset with GBDT predictions was split again into training and test data. The second layer of the proposed models with GBDT base contain GBDT, RF and DL models respectively, which were trained on the training data and tuned to get the best fits. We named these models GBDT-GBDT, GBDT-RF, and GBDT-DL Stacked Models. The Layer 2 models' predictions on the test dataset were used to evaluate the proposed models.

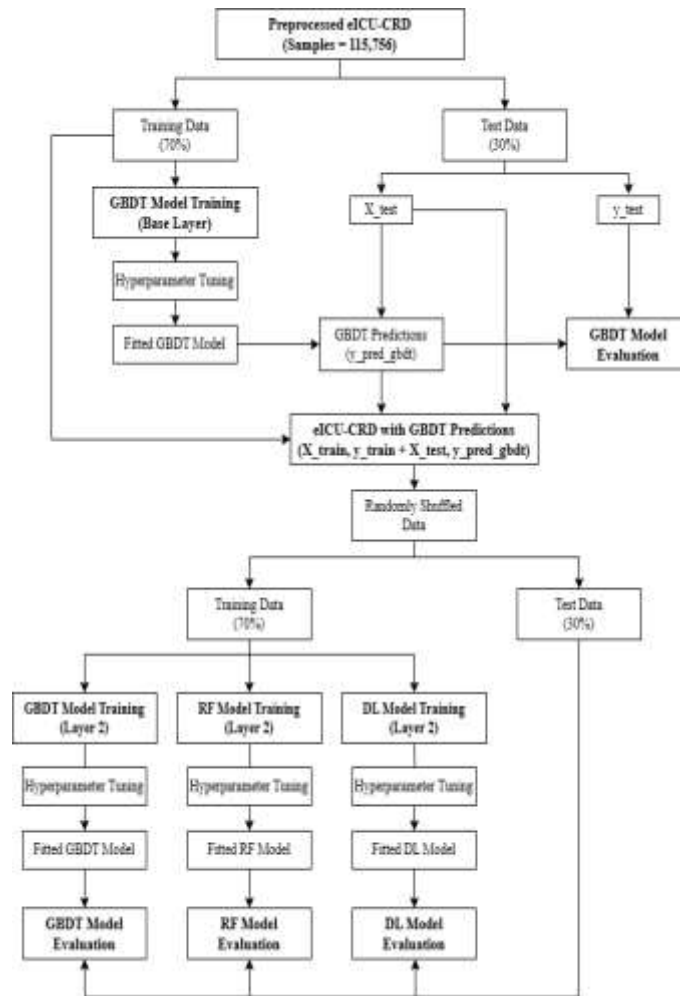


Fig. 2. Development of stacked models with GBDT base

4.4.3. Stacked Models with RF Base

Figure 3 illustrates the process of developing the three stacked models with the RF model as base. The saved dataset with RF predictions was split again into training and test data. The second layer of the proposed models with RF base contain GBDT, RF and DL models respectively, which were trained on the training data and tuned to get the best fits. We named these models RF-GBDT, RF-RF, and RF-DL Stacked Models. The Layer 2 models' predictions on the test dataset were used to evaluate the proposed models.

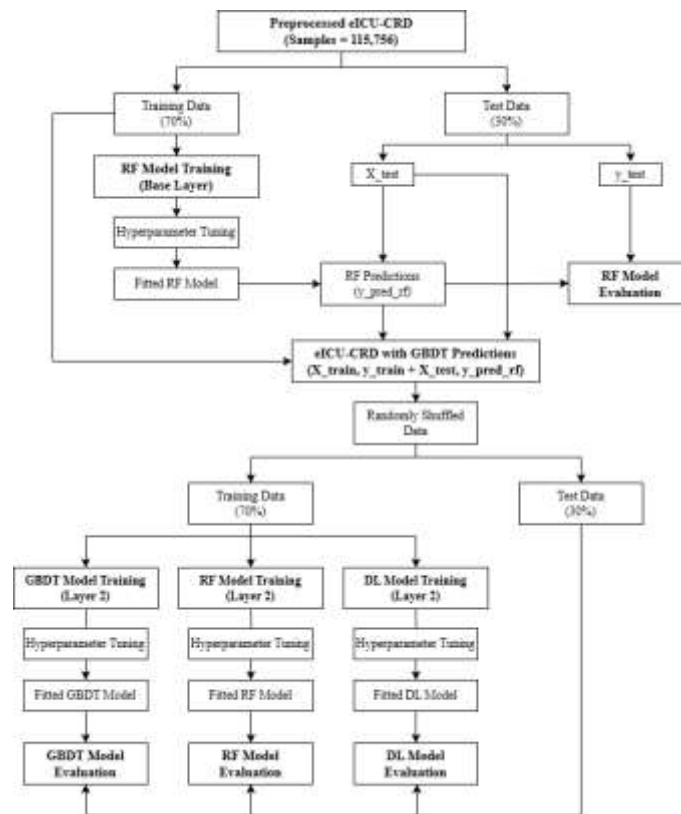


Fig. 3. Development of stacked models with RF base

4.4.4. Stacked Models with DL Base

Figure 4 illustrates the process of developing the three stacked models with the DL model as base. The saved dataset with DL predictions was split again into training and test data. The second layer of the proposed models with DL base contain GBDT, RF and DL models respectively, which were trained on the training data and tuned to get the best fits. We named these models DL-GBDT, DL-RF, and DL-DL Stacked Models. The Layer 2 models' predictions on the test dataset were used to evaluate the proposed models.

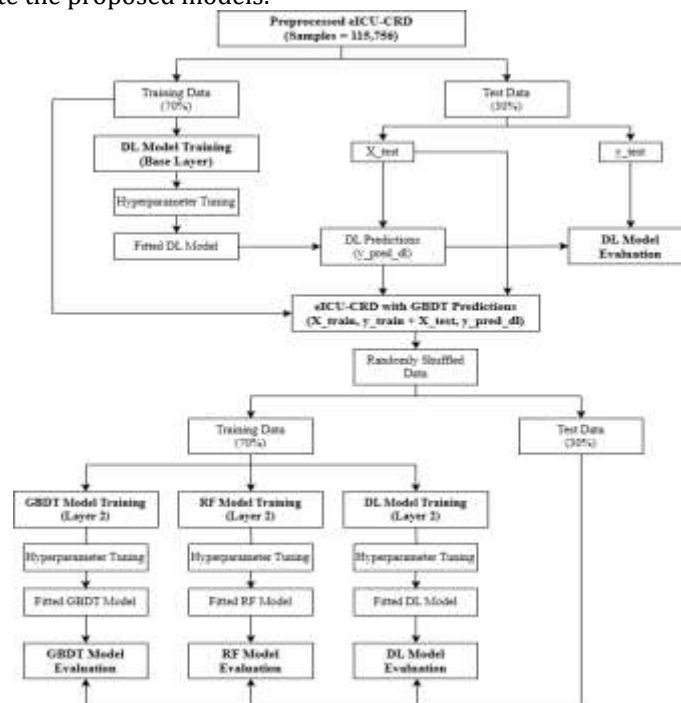


Fig. 4. Development of stacked models with DL base

5. Experiment and Results

5.1. Experimental Setup

The experimental arrangement contains Visual Studio Code (VS Code), one of the most popular software for code editing, which is used to write, edit and compile Python (version 3.8.10) code in which the machine learning classification models were implemented. The models are developed using widely-used Python libraries like Numpy and Pandas for data processing, Scikit-learn for ML models, and Matplotlib for data visualization. The eICU-CRD dataset is used to train and evaluate the prediction models. The hardware used for the task includes 64-bit Windows 11 OS, 12th Gen Intel(R) Core(TM) i5 processor, 16GB RAM and 512GB SSD storage.

5.2. Experiment

This study emphasises the implementation of prediction models for pLOS-ICU using data of general ICU patients from a multi-centre dataset, eICU-CRD. The eICU-CRD dataset initially comprised 200,859 ICU admission samples of 139,367 different patients. Out of these samples only 115,756 were left after data preprocessing. These 115,756 ICU admission samples were randomly split into a training set containing 81,029 samples and a test set with the remaining 34,727 samples. The GBDT, RF, and DL base layer models were trained on this training set, and their evaluation was done on the test set. The ICU-LOS column in the test set was replaced by the predictions of the base layer models and then combined with the training set to get three new datasets with predictions. These new datasets were then again randomly split into training and test sets which were used to train the layer 2 models, which we called GBDT-GBDT, GBDT-RF, GBDT-DL, RF-GBDT, RF-RF, RF-DL, DL-GBDT, DL-RF, and DL-DL, where the first model name indicates the base layer model and the second model name indicates the model used in layer 2. For example, GBDT-RF indicates that GBDT is the base layer model and RF is the layer 2 model.

5.3. Results

The statistical analysis of the dataset containing 115,756 samples, used to develop the models, yields the following characteristics of the dataset: The mean and median age of patients are 62 and 64 years respectively with a standard deviation of 16.5. The number of males and females in the dataset are 63,377 and 52,379 respectively. The mean and median values of LOS-ICU of patients in the dataset are 3 days and 1.85 days respectively, with a standard deviation of 3.99 days. The percentage of patients having pLOS-ICU is 28.64%. These characteristics are summarized in Table 2.

Table 2. Statistical analysis of processed eICU-CRD dataset

Patient Demographic	Value
Total Samples	115,756
No. of Males	63,377
No. of Females	52,379
Age (in years)	62 ± 16.5
LOS-ICU (in days)	3 ± 3.99
pLOS-ICU (in %)	28.64

A total of nine prediction models for pLOS-ICU are developed in this study, where all of the proposed models have two layers of models. The models were evaluated using three metrics, namely AUROC, AUPRC, and Brier score. The performance of all the nine proposed models along with the three base models is summarized in Table 3.

Table 3. Performance of proposed pLOS-ICU prediction models on the eICU-CRD dataset

Model	AUPRC	AUROC	Brier Score
GBDT (Base)	0.742	0.561	0.172
RF (Base)	0.732	0.544	0.175
DL (Base)	0.731	0.546	0.176
GBDT-GBDT	0.783	0.595	0.140
GBDT-RF	0.780	0.589	0.143
GBDT-DL	0.777	0.580	0.143
RF-GBDT	0.780	0.586	0.143
RF-RF	0.778	0.586	0.143
RF-DL	0.774	0.570	0.145
DL-GBDT	0.774	0.574	0.142

Model	AUPRC	AUROC	Brier Score
DL-RF	0.772	0.572	0.143
DL-DL	0.772	0.573	0.142

Table 3 implies that GBDT-GBDT is the model with the best performance out of all the proposed models with respect to AUROC, AUPRC and Brier score. We can also see that all the proposed models outperform the three base models. The nine prediction models for pLOS-ICU are also compared on the basis of their calibration curves. The curves are represented in Figure 5 for all the proposed models.

The calibration curves in Figure 5 imply that the RF-GBDT model is the best calibrated model, while DL-GBDT and GBDT-GBDT models are also not very far from the RF-GBDT model in terms of calibration.

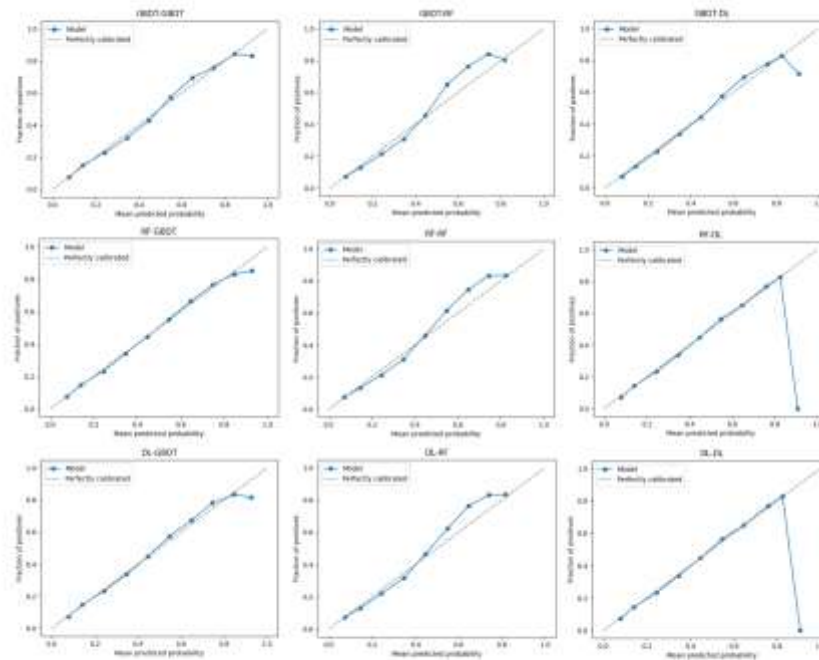


Fig. 5. Calibration Curves of the nine proposed models

6. Discussion

The purpose of this study was to develop and validate enhanced ML prediction models for predicting pLOS-ICU for general ICU patients using the eICU-CRD dataset. A total of nine double layered stacked models were proposed in this study to improve the performance achieved by single layer ML models. The outcomes of this study demonstrate the ability of combination of different models in the two-layered model framework. Among the nine developed models, namely GBDT-GBDT, GBDT-RF, GBDT-DL, RF-GBDT, RF-RF, RF-DL, DL-GBDT, DL-RF, and DL-DL, the top three best-calibrated models were RF-GBDT, DL-GBDT, and GBDT-GBDT. This implies that these models correspond with our results and show high reliability in their predictions.

Particularly, the RF-GBDT model, which uses an RF base model followed by a GBDT model in the second layer, performed the best in terms of calibration. This suggests that RF as a base model effectively identifies diverse patterns in the data and GBDT in the second layer adds further improvement to predictions. Also, all three best-calibrated models contain a GBDT model in the second layer, which suggests that GBDT is effective in refining probability estimates and improving model calibration in the second stage of the prediction framework. This is likely because GBDT models correct errors sequentially, reduce bias, and generate well-calibrated outputs. When GBDT models are placed in the second layer, they might help reduce miscalibrations of the base model and improve the reliability of predictions.

Furthermore, the models with the best performance based on AUROC, AUPRC, and Brier score were: 1) GBDT-GBDT, which achieves the best balance between overall accuracy and calibration; 2) GBDT-RF, which utilizes the predictive strength of the GBDT model in the base layer and takes advantage of the robustness of the RF model in the second layer; and 3) RF-GBDT, which demonstrates strong performance across different metrics including calibration, further supporting the effectiveness of RF as a base model and GBDT as a second-layer model. The performance of the GBDT-

GBDT model in both calibration and accuracy metrics suggests that when boosting-based models are used in both layers, they tend to be more effective. The strong performance of RF-GBDT and GBDT-RF models further suggests that the combination of ensemble methods, particularly the complementing strengths of RF and GBDT, helps improve feature learning and generalization [59, 60].

Conversely, the three worst-calibrated models were DL-DL, RF-DL, and GBDT-RF, while the models with the worst performance in terms of AUROC, AUPRC, and Brier score were DL-DL, DL-RF, and RF-DL. These results clearly show that DL in the second layer consistently leads to poor performance and weak calibration in comparison to GBDT and RF [61, 62]. This could be because deep learning models need vast amounts of data to generalize well, and in the stacked approach used in this study, the second-layer model receives some prediction inputs from the base model rather than raw data. Additionally, it is well known that neural networks generate overconfident probability estimates, which can lead to poor calibration if not properly corrected. The DL-DL model performed the worst across all metrics, suggesting that using DL as both the base and second-layer model may increase errors or introduce instability in probability estimation. Also, the poor calibration of GBDT-RF suggests that RF may not be as effective as GBDT in refining probability outputs when placed in the second layer. Unlike GBDT, RF does not iteratively correct prediction errors, which might explain its lower calibration performance in this approach.

Overall, these findings emphasize that GBDT might be the most effective second-layer model for improving calibration, while deep learning struggles when used in this role. Future research could consider using additional strategies such as alternative stacking techniques to further improve predictions.

The analysis of feature importance of the best performing models, namely GBDT-GBDT and RF-GBDT, reveals the major predictors which influence the prediction process of these two models. Both models identified the same top five most important features: GCS, PaO₂/FiO₂ ratio, minimum systolic blood pressure, heart rate, and blood urea nitrogen (BUN). Out of these five features, GCS was the most important predictor, contributing 23.9% to the GBDT-GBDT model and 23% to the RF-GBDT model. GCS, which is a measure of the neurological function of the body, is often used as an indicator of the severity of a patient's condition, especially in the intensive care unit. Its high importance in the prediction of pLOS-ICU suggests that the neurological status of a person plays an important role in patient outcomes, which aligns with existing clinical knowledge [63, 64].

The PaO₂/FiO₂ ratio, a key measure of respiratory function and oxygenation efficiency, was the second most important feature, contributing 19.3% and 18.6% in the GBDT-GBDT and RF-GBDT models respectively. This highlights how respiratory function plays an important role in predicting the declining health of a patient [65]. The significant weight of this feature suggests that oxygenation status is a major determinant of risk in this predictive framework [66]. Minimum systolic blood pressure, which contributed 8.5% in GBDT-GBDT and 8.8% in RF-GBDT, and heart rate, which contributed 8.3% in GBDT-GBDT and 8.5% in RF-GBDT, also ranked highly, showing the importance of hemodynamic stability in patient outcomes. The presence of these features shows the importance of circulatory health for the prediction of pLOS-ICU [67]. Lastly, BUN (blood urea nitrogen, also known as serum urea nitrogen) contributed 7% in the GBDT-GBDT model and 7.3% in the RF-GBDT model. This indicates that renal function and metabolic status are also important factors in predicting pLOS-ICU. Elevated BUN levels can indicate kidney dysfunction, dehydration, or increased catabolic activity, all of which are associated with poor prognosis in critically ill patients [68].

The consistency in the ranking of all these features across both models suggests that these features are effective and reliable predictors, which supports their relevance in the healthcare industry. The strong performance of GBDT-GBDT and RF-GBDT models in their prediction accuracy and calibration may be partially associated with their ability to effectively use these important features.

7. Conclusion and Future Work

In conclusion, we proposed nine stacked machine learning models for the prediction of pLOS-ICU in this study. Out of all the nine proposed models, the GBDT-GBDT model outperforms all the other eight models for the prediction of pLOS-ICU. This prediction model might be used to aid ICU clinicians in determining patients with a potential risk of pLOS-ICU, helping them take necessary decisions for the health of patients and the optimal management of hospital resources. The models proposed in this study for pLOS-ICU prediction may be further improved using more sophisticated machine learning techniques for use in the health sector. Other predictor variables may also be identified which prove to be more correlated to pLOS-ICU. Models can also be developed for other patient-related aspects like disease detection, mortality prediction, etc., or hospital-related aspects like resource usage prediction and management, medicine usage prediction, etc., which can help clinicians make decisions faster to provide better healthcare facilities to patients and use their resources optimally.

References

1. Scott S, Millar BC, McClean S. Editorial: Education and Training in Biomedical Science. *Br J Biomed Sci*. 2024 Sep 2;81:13598. Doi: 10.3389/bjbs.2024.13598. PMID: 39285897; PMCID: PMC11402700.
2. Huang Y., et al. (2023). Deep Learning Prediction Model for Patient Survival Outcomes in Palliative Care Using Actigraphy Data and Clinical Information. *Cancers*, 15(8):2232. DOI: 10.3390/cancers15082232

3. Huang S., Chaudhary K., Garmire L.X. (2017). More Is Better: Recent Progress in Multi-Omics Data Integration Methods. *Front. Genet.*, 8:84. DOI: 10.3389/fgene.2017.00084
4. Carra G., et al. (2020). Data-driven ICU management: Using Big Data and algorithms to improve outcomes. *J. Crit. Care*, 60, 300–304. DOI: 10.1016/j.jcrc.2020.09.002
5. Subrahmanya S.V.G., et al. (2022). The role of data science in healthcare advancements: applications, benefits, and future prospects. *Ir. J. Med. Sci.*, 191(4), 1473–1483. DOI: 10.1007/s11845-021-02730-z
6. Celi L.A., et al. (2013). "Big data" in the intensive care unit. *Am. J. Respir. Crit. Care Med.*, 187(11), 1157–1160. DOI: 10.1164/rccm.201212-2311ED
7. Sanchez-Pinto L.N., et al. (2018). Big Data and Data Science in Critical Care. *Chest*, 154(5), 1239–1248. DOI: 10.1016/j.chest.2018.04.037
8. Seymour T., et al. (2014). Electronic Health Records (EHR). DOI: 10.19030/ajhs.v3i3.7139
9. Kim M.K., et al. (2024). Challenges in and Opportunities for EHR-Based Data Analysis and Interpretation. *Gut Liver*, 18(2), 201–208. DOI: 10.5009/gnl230272
10. Holmes J.H., et al. (2021). Why Is the EHR So Challenging for Research and Clinical Care?. *Methods Inf Med*, 60(1-02), 32–48. DOI: 10.1055/s-0041-1731784
11. Binson V.A., et al. (2024). A Review of Machine Learning Algorithms for Biomedical Applications. *Ann Biomed Eng*, 52, 1159–1183. DOI: 10.1007/s10439-024-03459-3
12. Cascarano A., et al. (2023). Machine and deep learning for longitudinal biomedical data. *Artif Intell Rev*, 56(Suppl 2), 1711–1771. DOI: 10.1007/s10462-023-10561-w
13. Chakraborty C., et al. (2023). From ML to DL: A data-driven paradigm shift in healthcare. *Curr Res Biotechnol*. DOI: 10.1016/j.crbiot.2023.100164
14. Baghel R., et al. (2023). Application of Machine Learning in Biomedical Field.
15. Rahman A., et al. (2024). ML and DL-based approach in smart healthcare. *AIMS Public Health*, 11(1), 58–109. DOI: 10.3934/publichealth.2024004
16. Wang Y., et al. (2023). Trends in using DL in biomedical prediction systems. *Front. Neurosci.*, 17:1256351. DOI: 10.3389/fnins.2023.1256351
17. Friedman J.H. (2001). Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5), 1189–1232. DOI: 10.1214/aos/1013203451
18. Ke G., et al. (2017). LightGBM: a highly efficient gradient boosting decision tree. *NIPS*, 3149–3157.
19. Friedman J.H. (2002). Stochastic Gradient Boosting. *Comput. Stat. Data Anal.*, 38, 367–378. DOI: 10.1016/S0167-9473(01)00065-2
20. Rokach L. (2010). Pattern Classification Using Ensemble Methods. DOI: 10.1142/7238
21. Pedregosa F., et al. (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.*, 12, 2825–2830.
22. Breiman L. (2001). Random Forests. *Mach. Learn.*, 45, 5–32. DOI: 10.1023/A:1010933404324
23. Geurts P., Ernst D., Wehenkel L. (2006). Extremely randomized trees. *Mach. Learn.*, 63(1), 3–42. DOI: 10.1007/s10994-006-6226-1
24. James G., et al. (2023). An Introduction to Statistical Learning with Applications in Python. Springer. DOI: 10.1007/978-3-031-38747-0
25. Sarker I.H. (2021). Deep Learning: A Comprehensive Overview. *SN Comput. Sci.*, 2, 420. DOI: 10.1007/s42979-021-00815-1
26. Alzubaidi L., et al. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications. *J Big Data*, 8, 53. DOI: 10.1186/s40537-021-00444-8
27. Chollet F. (2017). Deep Learning with Python. Manning Publications.
28. Uhrig R.E. (1995). Introduction to artificial neural networks. *IECON*, 33–37. DOI: 10.1109/IECON.1995.483329
29. Sonali M., Maind B., Wankar P. (2014). Research Paper on Basics of Artificial Neural Networks.
30. Glorot X., Bengio Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *AISTATS*.
31. Hinton G.E. (1990). Connectionist learning procedures. *Machine Learning: Paradigms and Methods*, Elsevier, 185–234.
32. McClish D.K. (1989). Analyzing a Portion of the ROC Curve. *Med. Decis. Making*, 9(3), 190–195. DOI: 10.1177/0272989X8900900307
33. Fawcett T. (2006). An introduction to ROC analysis. *Pattern Recognit. Lett.*, 27, 861–874. DOI: 10.1016/j.patrec.2005.10.010
34. Hand D.J., Till R.J. (2001). A Simple Generalisation of the AUC for Multi-class Problems. *Mach. Learn.*, 45, 171–186. DOI: 10.1023/A:1010920819831
35. Seits F., et al. (2022). Evaluation of Color Anomaly Detection in Multispectral Images. *Eng*, 3(4), 541–553. DOI: 10.3390/eng3040038
36. Brier G.W. (1950). Verification of Forecasts Expressed in Probability Terms. *Mon. Weather Rev.*, 78, 1–3. DOI: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2
37. Rufibach K. (2010). Use of Brier score to assess binary predictions. *J. Clin. Epidemiol.*, 63(8), 938–939. DOI: 10.1016/j.jclinepi.2009.11.009

38. Herman C., et al. (2009). Predicting ICU LOS in CABG patients. *Interact. Cardiovasc. Thorac. Surg.*, 9(4), 654–658. DOI: 10.1510/icvts.2008.199521
39. Qian Q., et al. (2021). Prediction Models for AKI in ICU. *Int. J. Gen. Med.*, 14, 623–632. DOI: 10.2147/IJGM.S289671
40. Zoller B., et al. (2014). ICU LOS biomarkers study. *Ann. Intensive Care*, 4, 19. DOI: 10.1186/s13613-014-0019-7
41. Houthooft R., et al. (2015). Survival and LOS prediction in ICU. *Artif. Intell. Med.*, 63, 191–207. DOI: 10.1016/j.artmed.2014.12.009
42. Rotar E.P., et al. (2022). ICU LOS after cardiac surgery. *Semin. Thorac. Cardiovasc. Surg.*, 34(1), 172–179. DOI: 10.1053/j.semtcvs.2021.02.021
43. Iwase S., et al. (2022). ICU mortality and LOS prediction using ML. *Sci. Rep.*, 12, 12912. DOI: 10.1038/s41598-022-17091-5
44. Lin K., et al. (2019). AKI mortality prediction in ICU. *Int. J. Med. Inform.*, 125, 55–61. DOI: 10.1016/j.ijmedinf.2019.02.002
45. Meiring C., et al. (2018). Optimal ICU outcome prediction. *PLOS ONE*, 13(11): e0206862. DOI: 10.1371/journal.pone.0206862
46. Viton F., et al. (2020). Multi-Channel Convnet for ICU mortality risk.
47. Chen J., et al. (2023). DL for inpatient LOS and mortality. *J. Biomed. Inform.*, 147, 104526. DOI: 10.1016/j.jbi.2023.104526
48. Navaz A.N., et al. (2016). Data mining in ICU. *IIT*, 1–5. DOI: 10.1109/INNOVATIONS.2016.7880045
49. Rocheteau E., et al. (2020). TPCNs for ICU LOS. *Health, Inference, and Learning*. DOI: 10.1145/3450439.3451860
50. Ma X., et al. (2020). ICU LOS prediction with individual classifier. *Comput. Methods Programs Biomed.*, 186, 105224. DOI: 10.1016/j.cmpb.2019.105224
51. Hachesu P.R., et al. (2013). LOS prediction using data mining. *Healthc. Inform. Res.*, 19(2), 121–129. DOI: 10.4258/hir.2013.19.2.121
52. Samaga D., et al. (2020). Single vs multi-center data in prognostic modeling. *Radiat. Oncol.*, 15, 109. DOI: 10.1186/s13014-020-01543-1
53. Jo D. (2020). Interpretation bias in multicenter studies. *Korean J. Pain*, 33(3), 199–200. DOI: 10.3344/kjp.2020.33.3.199
54. Wu J., et al. (2021). Predicting prolonged ICU stay using ML. *Diagnostics*, 11, 2242. DOI: 10.3390/diagnostics11122242
55. Pollard T.J., et al. (2018). The eICU Collaborative Research Database. *Sci. Data*. DOI: 10.1038/sdata.2018.178
56. US Congress. (1996). Health Insurance Portability and Accountability Act. <https://www.govinfo.gov/app/details/PLAW-104publ191>
57. Le Gall J., et al. (1993). SAPS II: A new ICU score. *JAMA*, 270(24), 2957–2963. DOI: 10.1001/jama.1993.03510240069035
58. Verburg I.W.M., et al. (2014). Regression Methods for ICU LOS. *PLOS ONE*, 9(10): e109684. DOI: 10.1371/journal.pone.0109684
59. Delgado-Panadero A., et al. (2023). Distributed Gradient Boosting Forest. *Appl. Intell.*, 53, 22991–23003. DOI: 10.48550/arXiv.2402.03386
60. Zhao C., et al. (2020). BoostTree and BoostForest. *IEEE TPAMI*, 45, 8110–8126. DOI: 10.48550/arXiv.2003.09737
61. Unal A.F., et al. (2021). ML algorithms on telemetry data. *FiCloud*, 79–85. DOI: 10.1109/FiCloud49777.2021.00019
62. Zhang C., et al. (2017). Classification algorithm comparison. *Expert Syst. Appl.* DOI: 10.1016/j.eswa.2017.04.003
63. Mehta R., Chinthapalli K. (2019). Glasgow coma scale explained. *BMJ*, 365, l1296. DOI: 10.1136/bmj.l1296
64. Jain S., Iverson L.M. (2023). Glasgow Coma Scale. In: *StatPearls*. StatPearls Publishing.
65. El-Khatib M.F., Jamaledine G.W. (2008). PaO₂/FiO₂ ratio relevance. *Crit. Care*, 12(1), 407. DOI: 10.1186/cc6777
66. Esteve F., et al. (2014). PaO₂/FiO₂ after cardiac surgery. *BMC Anesthesiol.*, 14, 83. DOI: 10.1186/1471-2253-14-83
67. Rahman A., et al. (2021). ML prediction of ICU interventions. *Crit. Care*, 25, 388. DOI: 10.1186/s13054-021-03808-x
68. Lan Q., et al. (2021). BUN and cardiovascular risk. *Front. Cardiovasc. Med.*, 8, 614117. DOI: 10.3389/fcvm.2021.614117
69. Bernstam EV, Smith JW, Johnson TR. What is biomedical informatics? *J Biomed Inform.* 2010 Feb;43(1):104-10. doi: 10.1016/j.jbi.2009.08.006. Epub 2009 Aug 13. PMID: 19683067; PMCID: PMC2814957.