AIMS Mathematics

AIMS Mathematics, 9(4): 7751–7769.
DOI: 10.3934/math.2024376
Received: 23 December 2023
Revised: 30 January 2024
Accepted: 06 February 2024
Published: 23 February 2024

http://www.aimspress.com/journal/Math

Research article

Predicting hospital disposition for trauma patients: application of data-driven machine learning algorithms

Nasser Alrashidi[1,*], Musaed Alrashidi[2], Sara Mejahed[3] and Ahmed A. Eltahawi[3]

[1] Department of Surgery, College of Medicine, Qassim University, Buraydah, Qassim, Saudi Arabia
[2] Department of Electrical Engineering, College of Engineering, Qassim University, Buraydah 52571, Saudi Arabia; malrashidi@qu.edu.sa
[3] Information System Department, Faculty of Computers and Informatics, Suez Canal University, Egypt; sara.mongy@ci.suez.edu.eg, a.othman@ci.suez.edu.eg

* **Correspondence:** Email: n.alrashidi@qu.edu.sa.

Abstract: As a consequence of road accidents, around 1.3 million people die, and between 20 to 50 million have nonfatal injuries. Therefore, hospitals are receiving a high volume of patients in their urgent care, and a quick decision must be made regarding their treatment plans. At the admission stage, there is no information or probability about the patient's final result, regardless of if the patient will mostly die or be safely discharged from the hospital. To address this issue, this study proposed a machine learning-based framework that can predict the hospital disposition for trauma patients. The framework was developed to anticipate whether the patient would be safely discharged from the hospital or die based on a set of features collected at the admission time. In this study, the data used was collected from the King Abdulaziz Medical City (KAMC) in Riyadh, Saudi Arabia, and the performance of different machine learning algorithms was investigated, including eXtreme gradient boost (XGBoost), K-nearest neighbor, random forest, logistic regression, BRR, and support vector machine. Results show that the XGBoost algorithm demonstrated a high degree of detection and prediction accuracy for disposed-to-home patients; of the 6059 patients that were sent home, the XGBoost correctly predicted 5944 (98%) of the total. Finally, the developed framework could accurately predict hospital disposition for trauma patients with high accuracy and sensitivity levels. This system can benefit healthcare teams and insurance companies by providing them with a quick decision-making tool to determine the best treatment plan for patients.

Keywords: health care; machine learning; time series forecasting; data mining; classification

algorithms; bias-variance trade-off
Mathematics Subject Classification: 68Q30

Abbreviations

The following abbreviations are used in this manuscript: ICU: intensive care unit; ML: machine learning; XGBoost: eXtreme gradient boost; KNN: K-Nearest neighbor; RF: random forest; LG: logistic regression; BRR: Bayesian ridge regression; SVM: support vector machine; AI: artificial intelligence; ISS: injury severity score; BP: blood pressure; HR: heart rate; EDNN: ensemble of deep neural networks; AUC: area under the curve; TBI: traumatic brain injury; GCS: Glasgow coma scale scores; SBP: systolic blood pressure; ED: emergency department; KAMC: King Abdulaziz Medical City; MAD: median absolute deviation; P: precision

1. Introduction

The disposition of trauma patients after initial stabilization in the emergency department is a crucial aspect of their management [1]. The proper hospital disposition is vital for these patients to have the most outstanding results possible. The disposition decision involves determining whether the patient can be admitted to a regular hospital floor, discharged to their home, transferred to a higher level of care, or admitted to an intensive care unit (ICU) [2,3]. This decision is based on several factors, including the severity and mechanism of injury, the patient's physiological status, and the availability of appropriate resources at the receiving facility [4]. Trauma is a leading cause of morbidity and mortality worldwide, with an estimated 5 million deaths annually. In the United States, trauma is the leading cause of death in individuals aged 1–44 years [5]. The trauma system is designed to provide comprehensive care to injured patients, from prehospital care to definitive treatment and rehabilitation. The American College of Surgeons Committee on Trauma has established guidelines for the optimal management of trauma patients, including hospital disposition decisions [6]. The disposition decision for trauma patients has significant implications for healthcare utilization and costs. Inappropriate disposition decisions can lead to unnecessary hospitalizations, increased healthcare costs, and potential adverse outcomes for the patient [7]. Conversely, early identification of patients who require a higher level of care can facilitate appropriate triage and transfer, thus leading to improved outcomes. Therefore, it is essential to identify factors that predict the need for a higher level of care and to develop protocols and guidelines for appropriate disposition decisions [8,9]. The new rise in machine learning (ML) techniques has produced encouraging outcomes in improving the prediction accuracy in various contexts and patient circumstances [10,11]. In comparison to classical statistics, modern ML techniques can reveal new patterns for non-linear, high-order interactions between independent variables and produce more reliable predictions [8]. Hence, the primary objective of this research work is to develop precise ML models, with a focus on predicting high-patient disposition strategies for trauma patients. The task at hand is to build predictive algorithms that can use a variety of patient data sources to forecast outcomes such as discharge. There are various methods available in the literature that assist in predicting a trauma patients' hospital death or discharge. However, the primary contributions of this study compared to other studies are as follows:

- To develop reliable ML models for high-patient disposition planning for trauma patients. The prediction models can provide healthcare professionals, patients, and their families with better care and knowledge for deciding on and planning discharge recommendations.

- To examine the performance of six ML prediction algorithms to exploit hidden patterns in large data and anticipate the future number of hospital dispositions. The MLs are eXtreme gradient boost (XGBoost), K-nearest neighbor (KNN), random forest (RF), logistic regression (LG), Bayesian ridge regression (BRR), and support vector machine (SVM).
- Although the developed framework aims to predict the number of hospital dispositions, the proposed framework can help predict other healthcare outcomes.

The rest of the study is structured as follows: Section 2 discusses and elaborates on various types of ML methods; a thorough overview of the literature on the use of ML in trauma incidents can be found in Section 3; the problem statement and the framework used to perform this study are described in Section 4; a comprehensive discussion of the data preparation techniques used is given in Sections 5 and 6; and Section 7 presents and discusses the primary results. Section 8 summarizes the study's conclusions.

2. Background review

In this paper, various ML methods are introduced to develop a predictive model of hospital discharge. ML is a branch of artificial intelligence (AI) that uses statistical algorithms to allow computers to learn from data, identify patterns, and make predictions or decisions. ML techniques can be broadly classified into two categories: supervised and unsupervised learning. Supervised learning trains models on labeled data to predict or classify new data using popular algorithms such as RF and SVM. In contrast, unsupervised learning uses unlabeled data to discover hidden patterns or structures, thereby employing techniques such as clustering, principal component analysis, and autoencoders. The research paper utilized supervised learning methods to develop their predictive model, and this section provides a comprehensive overview of the supervised ML techniques used in the study.

2.1. XGBoost

XGBoost is a commonly used framework for improving the prediction accuracy in regression and classification tasks. It achieves this through a combination of gradient-boosting methods and an ensemble of decision trees. It employs a recursive strategy to incorporate models until the desired performance metrics are achieved, thus making it an advanced version of tree gradient boosting algorithms [12]. XGBoost is highly efficient and scalable, thus allowing for the optimization of memory and hardware resources to handle large-scale models. Additionally, it can effectively handle sparse data and uses a weighted quantile sketch for approximate learning [13].

2.2. KNN

The KNN algorithm is a simple yet effective non-parametric classification and regression technique in ML [14]. It works based on similarity, thereby evaluating a data point's neighbors in the feature space to determine its classification. In its simplest form, KNN determines the class of an unclassified sample by comparing it to the nearest neighboring classes. The number of neighbors considered is indicated by the "K" in KNN, and the method finds the K nearest points by calculating lengths (often using Euclidean or other distance metrics) [15]. A majority vote among these neighbors is required for classification, and the unclassified data point is given to the class label that is most common among them. To forecast a continuous value in the regression, the values of the KNN are averaged. Because of its ease of interpretation, KNN is a simple technique; nevertheless, because it relies on selecting a distance measure and a suitable value for K, thus its implementation requires careful consideration to strike a balance between accuracy and complexity.

2.3. RF

RF is a popular ensemble learning method that is based on decision trees [16]. It combines multiple decision trees to improve the accuracy of predictions for new data and prevent overfitting [17]. It can handle high-dimensional data and is commonly used for classification, regression, and feature selection tasks. It is a more advanced version of decision trees and can be seen as an improvement to bagged decision trees [18]. The algorithm builds each decision tree by randomly selecting a subset of features and samples from the training data. The final prediction is made by aggregating the predictions of all the trees. RF has been found to perform well in various classification and regression problems.

2.4. LG

LG is one of the fundamental and often utilized statistical methods for binary classification applications in machine learning [19]. Though it appears like a regression algorithm, LG is actually a classification algorithm. It uses a logistic function suited to the data to describe the likelihood that an instance will belong to a specific category. Concerning the input features and the log odds of the result, LG assumes a linear connection. It maps the input properties through a sigmoidal function, which converts the output into a range between 0 and 1 to estimate the likelihood of an event occurring [20]. When this probability surpasses a predetermined threshold, the model designates the event as belonging to the positive class; in all other cases, it places it in the negative class.

2.5. BRR

The BRR regression technique blends a linear regression and the ideas of Bayesian inference [21]. Due to its probabilistic nature and assumption of a prior distribution over the regression coefficients, this model may deal with scenarios involving multiple predictors with insufficient data more skillfully. By assigning a prior probability distribution to the regression coefficients, BRR allows the model to include prior knowledge about the values of the coefficients. Typically, this distribution is Gaussian. In light of the observed data, BRR creates a posterior distribution over the coefficients using Bayesian inference to update these prior beliefs. The model makes predictions and assesses uncertainty using this posterior distribution, thus delivering not just point estimates, but also confidence intervals for the forecasts. Additionally, the BRR algorithm is utilized for classification problems [22].

2.6. SVM

SVM is a frequently used supervised learning algorithm for tasks such as classification, regression, and outlier detection [23]. Its primary objective is to locate the optimal hyperplane that can separate the data into different classes. The method operates by separating the training data set into distinct categories, thus maximizing the gap between them. When the novel data is evaluated, it is assigned to the nearest category according to some estimates [24]. For data that is linearly separable, the algorithm uses a hyperplane; for nonlinear data, it employs various types of kernel functions based on the type of data [25].

3. Related work

Trauma accidents are responsible for the highest number of fatalities and disabilities globally. These injuries occur due to external forces, such as falls, car accidents, and assaults. Although there have been advancements in trauma care, trauma-related injuries remain the leading cause of deaths worldwide [26]. Therefore, preventing trauma accidents is a critical public health priority, and multiple interventions have been implemented to minimize the frequency and severity of these incidents [27]. However, there is an urgent requirement for an automated system that quickly predicts the severity of trauma and provides initial diagnoses to enable prompt medical decisions and an appropriate action with the patients upon their traumatic exposure. Therefore, ML adoption to predict disease outcomes has become increasingly popular, with studies showing that ML models outperform traditional methods [28]. Hence, the autonomous improvement of modeling algorithms by ML has caused the wide use of ML in medicine. Specifically, ML has demonstrated promising findings in medical and emergency services, thus leading to positive outcomes in pre-hospital care, disease screening, clinical decision-making, and mobile health. Therefore, this section discusses the previous studies on using ML to predict the hospital discharge status of trauma patients.

Li et al. [29] proposed a model based on ML to estimate the probability of acute traumatic coagulopathy in trauma patients upon immediate hospital admission. The study used information from 1087 trauma patients admitted to a trauma facility in China from January 2013 to December 2018. To train and evaluate their ML model, the researchers gathered 21 clinical and laboratory parameters, such as age, sex, injury severity score (ISS), blood pressure (BP), and lab data such as platelet count and prothrombin time. Two distinct algorithms, RF and LG, were employed to create the predictive models. The models were evaluated based on various metrics such as accuracy, sensitivity, specificity, and the area under the receiver operating characteristic (ROC) curve. The RF model demonstrated superior results compared to the alternative model.

Lee et al. [30] introduced a new model, called an ensemble of deep neural networks (EDNN) model, to utilize AI techniques to forecast the in-hospital mortality of patients who suffered physical trauma. The authors proposed a novel approach to predict in-hospital mortality using a deep neural network. They utilized a nationwide population-based dataset of physical trauma patients in Korea to develop and validate their model. They used ML algorithms, including adaptive boosting, XGBoost, and neural network models to identify the crucial predictors of in-hospital mortality. The authors reported that their EDNN model achieved a high accuracy in predicting in-hospital mortality, with an area under the curve (AUC) of 0.894. Additionally, the authors observed that their model surpassed traditional methods in terms of accuracy.

Hsu et al. [31] developed an ML model to anticipate in-hospital mortality among Taiwan's traumatic brain injury (TBI) patients, thereby utilizing clinical measures and demographics. The study revealed that Glasgow coma scale scores (GCS), ISS, and systolic blood pressure (SBP) at emergency department (ED) admissions were the most significant predictors of in-hospital mortality. Additionally, the study identified efficient cutoff values for clinical measures to forecast mortality. The findings from this study can assist in clinical decision-making and the establishment of care-delivery protocols for high-risk TBI patients. The decision tree algorithms are precise in predicting the prognosis of TBI patients and supporting health professionals in evaluating and providing intensive care.

Wang et al. [32] aimed to evaluate the effectiveness of using XGBoost to predict mortality among TBI patients with a GCS score below 13. The study involved 368 patients, split into training and test sets. Results revealed that non-survivors had lower GCS, higher ISS, lower platelets, albumin, and hemoglobin levels, and higher glucose and prothrombin time levels. The XGBoost algorithm was found to be more accurate in mortality prediction compared to the LG model, with GCS, prothrombin time, and glucose being the most significant features. The study implies that XGBoost could be a beneficial tool for physicians in assessing high-risk TBI patients' poor outcomes. Further research is required to validate these findings and explore other ML algorithms.

In another study [33], the authors aimed to develop an ML model capable of forecasting injury severity in car accident patients treated at Level 1 trauma centers in Korea. To achieve this, they utilized data from the Korea accident study database and employed LG, extreme XGBoost, and a multilayer perceptron model during the development process. Moreover, they implemented four different data-sampling methods to address the issue of imbalanced clinical datasets. The study discovered that the balanced XGBoost model was effective in classifying injury severity, with an AUC of 0.896 and an under-triage rate of 6.1%. The researchers noted that the Delta-V, age, and principal were significant factors in determining injury severity. Furthermore, they highlighted the importance of choosing an optimal injury severity prediction model that took varying motor vehicle crash conditions into account. The use of data-sampling techniques for class-imbalanced datasets was found to enhance the predictive capabilities of the model.

The study proposed by [34] aimed to predict the survival of trauma patients using LG, SVM, neural network, and trauma and ISS models, which were assessed based on accuracy, sensitivity, specificity, and AUC measures. The findings showed that all four models demonstrated high accuracy and sensitivity levels of over 97.5% and 98.6%, respectively. However, the neural network model exhibited the highest specificity of 51.5%, followed by the trauma and ISS, SVM, and LG models. The study concluded that the neural network model showed the highest balanced accuracy and predictive specificity in the test dataset, thus indicating its potential to predict the survival of trauma patients and improve clinical decision-making.

4. Proposed techniques

In this section, the problem statement and the detailed description of the framework used in this research work are discussed.

4.1. Problem statement

Accurately predicting a patient's hospital disposition is a critical challenge in healthcare that impacts treatment plans, resource distribution, and patient outcomes. Therefore, the objective of this study is to predict the probable course of a patients' care after their hospital stay ends by utilizing ML algorithms. The methodology used in this work aims to create predictive models by utilizing a wide range of patient demographic, clinical, and historical data. Additionally, the methodology aims to investigate and assess the predictive accuracy of ML algorithms, such as XGBoost, KNN, RF, LG, BRR, and SVR.

4.2. Study framework

Six ML algorithms were examined to enhance the accuracy of predicting hospital disposition for trauma patients. The overall framework that elaborates the proposed prediction models is shown in Figure 1. This framework can be used to predict other healthcare outcomes. Each procedure is described in detail in the subsections and sections that follow.

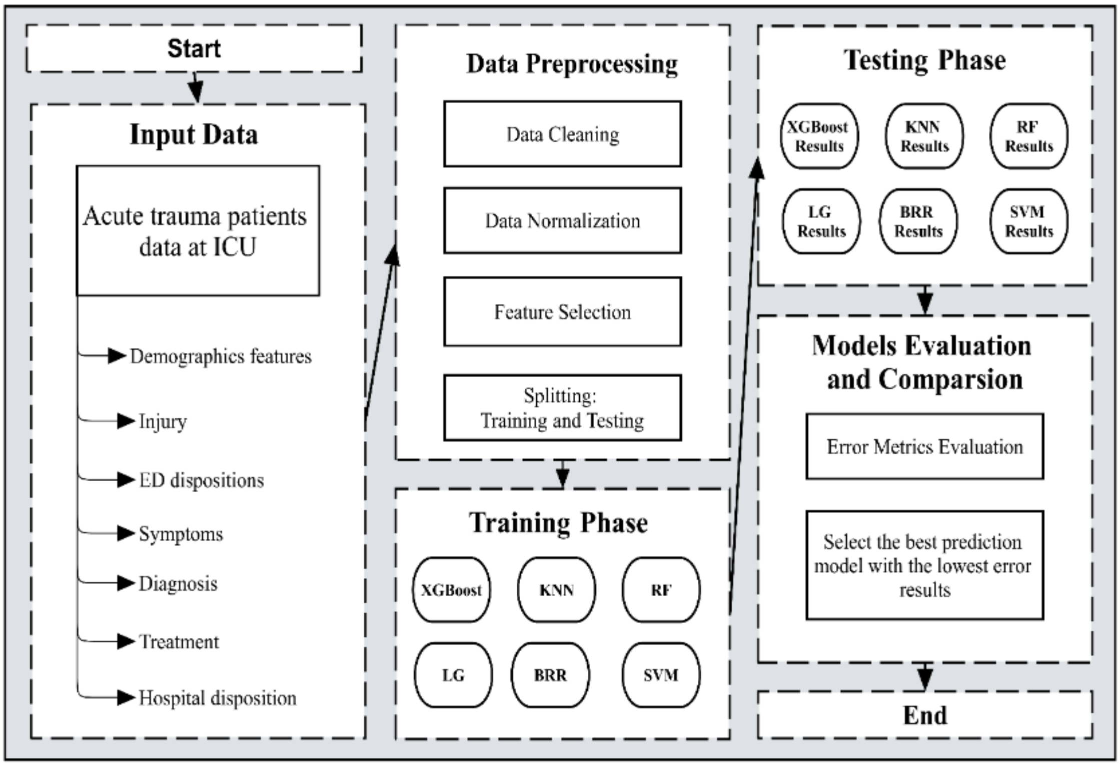

Figure 1. The framework of the proposed prediction models.

4.3. Data description

The database of King Abdulaziz Medical City (KAMC), Riyadh, Saudi Arabia, trauma was the source of the dataset used in this analysis. The database provided comprehensive information about all acute trauma patients who arrived at the hospital from January 2000 to January 2018. The KAMC is recognized as a specialized trauma center since it offers emergency care daily, thereby enabling access to healthcare plans for different services. Moreover, the KAMC is one of the few Saudi Arabian medical facilities recognized by the American College of Surgeons as offering specialist training in emergency life support. The database provides comprehensive data on trauma patients, including details on their age, sex, mode of transportation to the emergency department, vital signs when they arrived, and whether or not the trauma code was activated. The ISS and GCS were the severity scales utilized in this analysis. The analysis used these input variables for various outcomes, such as the length of stay in an ICU, the hospital length of stay, and hospital dispositions. The data consists of 22,212 patients with 19 different measures, including demographic features, injuries, ED dispositions, symptoms, diagnosis, treatment, and patient hospital dispositions. The features measures are summarized in Table 1.

Table 1. Summary of the categories and data included in the study.

Category	Data
Demographics	Age, weight, gender
Injury	Type of injury, mechanism of injury
Emergency department	Length of stay (hour), disposition
Symptoms	HR, RR, Sys. BP, GCS, ISS
Diagnosis and treatment	Diagnosis, ISS, procedure, trauma team activation
Hospital	Intubation days, ICU. days, hospital length of stay (days), hospital disposition

5. Data cleaning phase

A total of 22,212 patients with 19 different measures are available to predict the hospital disposition. A cleaning process is performed to prepare the data for analysis and prediction. The cleaning process proceeds as follows (see Algorithm 1):

- Three features were removed from the data, as they could not be used as predictors for forecasting the patient hospital disposition as they were available after the Emergency Department stage. These features are the incubation period (days), intensive care unit admission period (days), and hospital length of stay (days). The feature matrix now becomes of dimension 22,212×15.
- The target feature (patient hospital disposition) is filtered by removing any patient with a value not equal to death or home disposition. Three more values were found, namely "Transfer to another hospital" in 275 patients, "DOA" in 589 patients, and "Pending" in 174 patients. This step resulted in the discarding of 1038 patients, and the final feature matrix was reduced to 21,174×15, with 1113 (5% of the data) patients dying in the hospital and 20,061 (95% of the data) patients being disposed-to-home.
- Remove incomplete features. All features are processed, and if any features miss a value of more than 5% of the data, this feature will be removed from the data. This step resulted in removing the weight and procedure features, which were found missing in around 69% of patients who died in the hospital, and the feature matrix had a dimension of 21,174×13.
- Remove identical features. Any feature containing the same value will be removed from the data. This feature will not add any value to the training process as it lacks the necessary variability. To check if a feature has an exact value, the median absolute deviation (MAD) of each feature is calculated, and the feature is removed if its MAD is zero.
- Noise removal. For each patient, if any feature contains a missing value or noise value, such as inf or -inf, this patient will be removed. For the accuracy of the data, we preferred to remove patients with any missing or noisy data instead of using any missing imputation techniques. This step resulted in the discarding of 63 patients.
- Remove duplicates. In this step, for any two patients who have the same values with the same target value, one of them will be removed. This process resulted in removing one patient.
- Data pruning. In this step, a distance matrix using the Euclidean distance is calculated between all patients. If two patients are found to have the same features with different outputs, the one with the majority output will be removed. For example, if two patients have the same values for each feature, one disposed to home, and one died in the hospital; the one that was disposed to the home will be removed to try to increase the minority. This step is performed to avoid training confusion by feeding the machine learning with two exact data inputs and different outputs. No patient was removed as a result of this step, and the final feature matrix had a dimension of 21,111×13.

Algorithm 1. Data cleaning phase.

————**Load data**————
Load the data M= 22212×18.
————**Remove non-predictors**————
Remove three features not used in the prediction
F=22212×15
————**Target preparation**————
Set feature hospital disposition as target T
for any value V in T **do**
  if V not equal to home disposition or death in hospital **then**
    Remove patient with V
end if
end for
————**Filter features**————
for each column K in M **do**
  Calculate the missing percentage M_K
  Calculate the median absolute deviation D_K
  if M_K>n% OR D_K=0 **then**
    Remove K
  end if
end for
————**Noise removal**————
for each row R in M **do**
  if R contains missing or noise data **then**
    Remove R
  end if
end for
————**Pruning and duplication removal**——
Remove any duplicate in M
Calculate distance matrix D between all rows
if distance $D_{i,j}$ between row i and row j = 0 **then**
  if $T_i = T_J$ **then**
    Remove the row with the target dispose to the home
  end if
end if

5.1. Data analysis and statistics

Figure 2 summarizes the basic statistics (mean, standard deviation, and interquartile range (IQR)) for each column feature in the whole data and for each target category. The final feature matrix consists of 21,112 patients, with an average age of 28.62 and a standard deviation equal to 21.23. Around 95% of the patients were disposed from the hospital to home, while only 5% were reported as dead in the hospital. The available features are classified into numerical and non-numerical as follows:

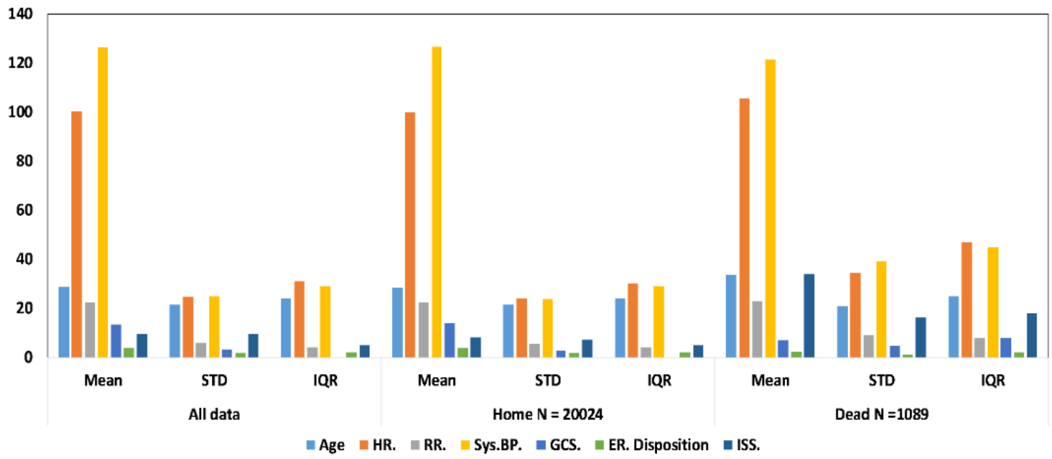

Figure 2. The statistical characteristic of the input features.

5.1.1. Numerical features: The data were collected on patients' admission to the hospital

1) Demographic data: The range of data, such as age, is between 0.1 to 110 years old, and the IQR=24.
2) Vital signs: Measurements calculated from the patient upon admission to the hospital, such as heart rate (HR), which is reported between 0 and 620 and IQR=31.
3) Systole blood pressure (Sys. BP): The range is between 0 and 1156, and IQR=29.
4) Respiratory rate (RR): The range is between 0 and 131, and IQR=4.
5) Symptoms: GCS is the most common scoring system for traumatized patients' consciousness levels. Its range is between 3 and 15 IQR=0.
6) ISS: An anatomical scoring system provides an overall score for patients with multiple traumas. The range is between 0 and 75 IQR=0.
7) ER Length of stay (hours): The range is between 0 and 939 hours.

5.1.2. Non-numerical data

1) Gender: around 23% of the patients were female, and 77% were found to be males. In patients disposed to their homes, 77% of the patients were male, and 85% of the patients who died in the hospital were also males.
2) Trauma team activation: reported as either yes or no and recorded as no in 85% of the patients. In the patients disposed to their homes, 87% were reported as no, while found in only 41% of the patients died in the hospital from those who reported yes.
3) Type of injury: It is categorized into 5 classes, namely ["BLUNT", "BURN/SCALD", "PENETRATING-Gunshot", "PENETRATING-OTHER", and "PENETRATING-Stab"], where BLUNT was found in 79% of the whole date, 83% of the disposed to home patients, and 84% of the patients who died in the hospital. The gunshot came in second place in the type of injury in 8% of the whole patients, 8% in patients disposed to home, and 11% of patients who died in the hospital.
4) Mechanism of injury: It is classified into 13 different classes. Falls came as the first mechanism, with 26% due to falling in the whole data, 27% in patients diagnosed at home, and 50% in the patients who died at the hospital.
5) ER Disposition: It is categorized into 5 classes, namely "BURN UNIT", "ICU", "OR", "OTHER", and "WARD". 62% of the patients have WARD value, and 65% of the patients disposed to home.

Approximately 38% of the patients who died at the hospital had an ER disposition value reported as the odd ratio (OR).

6) Diagnosis: There are around 36 different diagnoses found in the data. The main reasons reported are broken arm, broken femur, broken lower leg, burns, and head injury. Head injuries were found to be the main reason patients died in the hospital, with 59% of the data.

5.2. Feature selection

To prepare the feature matrix for training, a feature selection approach is applied to select the best features related to the target output. This process started by calculating the univariate LG between each feature and the target output. This process resulted in the following: (see Algorithm 2)

- Calculate the univariate LG between each feature and the target output.
- The Python function GLM.from_formula is used to establish the relationship. This function is imported from the statmodel.api package. As the target output is binary, the binomial model is used to build the logistic relationship.
- Calculate the OR, 95% confidence interval, and the P-value using univariate LG between each feature.
- Any feature with a P-value<0.05 is considered for the training process.
- Calculate the correlation between all features.
- If two features are found to be highly correlated, one of them is discarded.
- The feature with a lower standard deviation is selected to be discarded.
- The final feature matrix resulting from this process remains the same, and no features were discarded, as shown in Table 2.

Table 2. Univariate LG results.

Feature	OR	P-value
Type of injury	1.2	5.3E-0.6
Mechanism of injury	0.95	5.8E-05
Age	0.99	3.8E-15
Gender	0.57	1.1E-10
ER. Length of stay (hours)	1	1.8E-3
HR	1	1.5E-13
RR	0.99	1.7E-04
Sys. BP	1	1.1E-12
GCS	1.4	0
Trauma team activation	0.1	6.7E-270
ER. disposition	1.7	1.6E-198
Diagnosis	0.95	1.3E-59
ISS	0.8	0

Algorithm 2. Feature selection phase.

————**Load data**————
Load the data M=21111×13.
for any feature F in M, do
    Calculate the univariate relation between F and T
    Calculate the P-value P_F
    if $P_F<0.05$, **then**
        Select F in the final matrix
    else
        Discard F
    end if
end for
for any two features, $F1$ and $F2$, **do**
    Calculate the correlation C between $F1$ and $F2$
    if $C>0.95$, **then**
        Calculate standard deviation of $F1$ and $F2$ as $S1$ and $S2$
    if $S1>S2$ **then**
        Discard $F2$
    else
        Discard $F1$
        end if
    end if
end for
Save the final matrix $M1$

6. Training and testing phase

The final feature matrix resulting from the feature selection process is used as the input to the ML algorithms along with the target output. This process starts by randomly splitting the data into a training set (N\% of the data) and a testing test (100-N% of the data). In this study, the data is split into 70% as training and the remaining 30% as testing to evaluate the efficiency of the algorithm. Then, the training and the testing sets are normalized using a standard scaler to be ready for the training process. The normalized training feature is used as the input to the ML algorithm and the target output (the patient disposed at home or dead at the hospital) is used as the output. Then, the testing set is used as the input to the trained model and the predicted outputs are used in comparison with the available data to evaluate the performance of the algorithm, see Algorithm 3. Many ML techniques are employed in this experiment as follows:

- XGBoost: Implemented using the Python function *XGBClassifier*.
- KNN: Implemented using the Python function *KNNClassifier*.
- RF: Implemented using the Python function *RFClassifier*. We tested three versions of RF with different tuning parameters.
- LG: Implemented using the Python function *LG* with *Liblinear* as a solver.
- BRR: Implemented using the Python function *Bayesian ridge* imported from the linear model module.
- SVM: Implemented using the Python function *SVM. SVC* with probability set to true.

Algorithm 3 Training and testing phase.

————**Training phase**————
Load the data $M1$=21111×13
Load the target vector=2111×1
Randomly split $M1$ into Training TR and testing TS matrices
$MR=n\%*M1$
$MS=(100-N)\%*M1$
Split the target T into TR and TS for training and testing
Normalize MR and MS
for each Algorithm R, **do**
Train R using MR as input and TR as output
Save the trained algorithm R
end for
———— **Testing phase** ————
Load R, MS, TS
for each algorithm, R **do**
Test R using MS and input
Save the estimated output ER
Compare ER and TS and save the results
end for
Repeat the training/testing process N times

7. Experiments and results

7.1. Evaluation metrics

To evaluate the performance of the ML algorithms, the confusion matrix is calculated, and the following metrics are calculated:

- Overall accuracy: It counts how many of a model's predictions are accurate overall. The accuracy formula is as follows [31,35]:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}. \tag{1}$$

- Recall: It measures the algorithm performance per class. For example, if the algorithm predicts N patients as going home, then it is the correct percentage of N:

$$Recall = \frac{TP}{TP+FN}. \tag{2}$$

- Precision (P): It measures the algorithm's accuracy in predicting each class. For example, if we have N patient going home, then it is the percentage of N that the algorithm can predict:

$$Precision = \frac{TP}{TP+FP}. \tag{3}$$

- F1 score: It measures the balance between P and recall:

$$F1Score = \frac{2\times Recall\times Precision}{Recall+Precision}. \tag{4}$$

- AUC: It calculates that the area under the curve between the actual and the estimated outputs (see [36,37]). The Python function *roc_auc_score* was used to calculate the AUC.

Where *TP* is the true positive, *FP* is the false positive, *TN* is the true negative, and *FN* is the false negative.

7.2. Results

In this section, the prediction models are compared according to some criteria to examine their performance in predicting the hospital disposition of patients. Results of the best forecasting models according to *P*, recall, F1-score, accuracy, and the AUC are listed in Table 3. In addition, Figure 3 depicts the confusion matrix of the six ML algorithms considered in this study. These ML algorithms were chosen over other existing algorithms because of their superior performance in the prediction challenge at hand. The XGBoost technique uses an ensemble of decision trees, thus making it appropriate for a variety of dataset sizes with high-accuracy prediction results. KNN is simple to build and good for classification jobs, whereas the LG algorithm is highly effective in computing and is suitable for binary classification applications. On the other hand, BRR incorporates probabilistic modeling, which allows for an uncertainty estimation in forecasts. Finally, SVM has successfully dealt with complex decision boundaries, making it suitable for classification problems.

Table 3. Summary Results of the considered ML algorithms.

Model	Label	*P*	Recall	F1-Score	Accuracy	AUC
XGBoost	Home	98%±0.0005	99%±0.0006	0.99±0.0005	97%±0.0008	0.98±0.0037
	Dead	83%±0.0118	66%±0.0056	0.73±0.0064		
RF	Home	98%±0.0007	99%±0.0010	0.99±0.0008	97%±0.0015	0.96±0.0055
	Dead	84%±0.0197	65%±0.0116	0.73±0.0133		
LG	Home	97%±0.0005	99%±0.0005	0.98±0.0004	97%±0.0006	0.96±0.0037
	Dead	77%±0.0109	53%±0.0092	0.63±0.0083		
BR	Home	97%±0.0079	99%±0.0087	0.98±0.0008	96%±0.0015	0.96±0.0035
	Dead	73%±0.1102	51%±0.1615	0.57±0.0957		
SVM	Home	97%±0.0015	99%±0.0011	0.98±0.0009	97%±0.0016	0.90±0.0105
	Dead	82%±0.0218	50%±0.0208	0.62±0.0133		
KNN	Home	97%±0.0014	99%±0.0015	0.98±0.0009	96%±0.0017	0.84±0.0088
	Dead	74%±0.0202	50%±0.0189	0.59±0.0100		

In this study, five independent experiments were conducted for each of the six ML algorithms, creating a total of 30 different forecasting models. Table 3 summarizes the average results of five experiments of the training and testing process. The average accuracy and the standard deviation were recorded to evaluate the performance and stability of the algorithms.

The available data is very biased towards home disposition (95% of the data), which makes the prediction process very challenging. Therefore, we are reporting not only accuracy, but also P, recall, and AUC to evaluate the ability of the algorithms to predict within the minority class.

XGBoost came in first place, with the overall accuracy reaching 97% and a standard deviation close to zero for the five experiments. RF and LG had the same overall accuracy as well, and the prediction will be biased toward the majority class. Therefore, the overall accuracy is not a suitable metric for evaluating this problem. On the other hand, KNN showed the lowest accuracy results in predicting the hospital discharge of patients.

The five algorithms managed to predict disposition to home very accurately: XGBoost reached 98%, RF reached 98% but with a higher standard deviation, and LG reached 97% and came in third place. BRR reached 97%, while SVR *P* results were found to be 97% with 0.0015 standard deviations.

In terms of predicting the minority class, XGBoost came in first place, with *P* reaching 83% and standard deviation=0.0118 for the five experiments, its recall reaching 66%, and its AUC was the best with 0.98. RF had a slightly better *P* than XGBoost, but its stability was lower due to a larger standard deviation.

Figure 3 shows a confusion matrix of all the considered algorithms. It is clear that the XGBoost algorithm can easily detect and predict disposed-to-home patients: out of 6059 patients going home, the algorithm managed to successfully predict 5944 with a percentage reaching 98%. Moreover, the algorithm has a high detection accuracy of dead patients, as it correctly predicted 228 patients out of the 274 patients that died in the hospital. Overall, the XGBoost algorithm demonstrated a high degree of detection and a prediction accuracy for disposed-to-home patients. By utilizing the study's developed framework, medical teams would have an invaluable tool to help them decide on treatment plans and allot hospital resources.

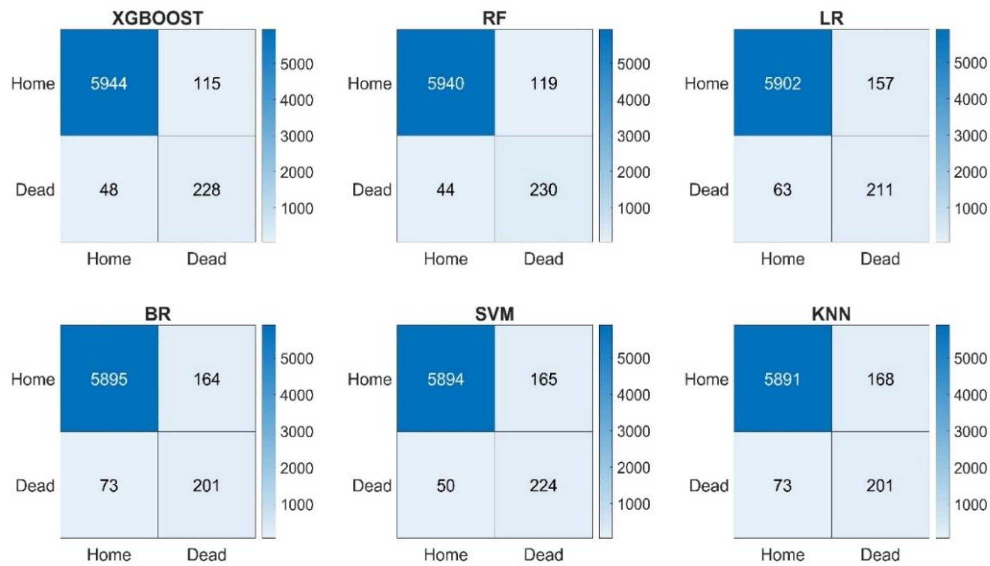

Figure 3. The confusion matrix results for the six ML algorithms.

7.3. Implications of findings and study limitations

The findings of the research have significant implications for healthcare administration. The ML algorithm developed in the current research managed to use many factors that are regularly collected in the emergency department to properly predict a patient's condition while they are in the hospital. With the aid of this method, clinical professionals and family members may find it easier to make fast, data-driven, and reliable treatment decisions. According to the analysis conducted in this study, there are a number of significant predictors that can lead to accurate prediction outcomes, including the GCS, ISS, SBP, HR, age, gender, and type of injury. These factors are easily evaluated in the emergency department and can be used to identify individuals who have a high mortality risk.

Additionally, it was discovered that the XGBoost algorithm was the best at forecasting a patients' hospital outcomes. In terms of classification, P, recall, F1 score, and AUC, XGBoost exhibited the highest average success rate. This can be attributed to the fact that the XGBoost technique employs an ensemble of decision trees, making it suitable for a wide range of dataset sizes while producing high-accuracy predictions. This algorithm's excellent accuracy raises the possibility that it could be applied to the creation of clinical decision support systems, which would enable medical professionals to make better treatment choices.

Regarding the potential limitations of this study, there are several research restrictions while developing a ML prediction model for hospital disposition. To begin, the integrity and accessibility of healthcare data provide hurdles, as errors, insufficient data, or prejudices in the dataset, which can all have an impact on the P of the prediction models. Second, it is vital to choose appropriate attributes, as omitting critical variables could affect the model's predictive ability. Furthermore, the model's generalizability to various healthcare environments and patient demographics may be hampered, thus limiting its wider application. Finally, the shifting nature of healthcare procedures, legal issues over the confidentiality of patient information, and the interpretability of complicated models all add to the potential constraints. Hence, overcoming these challenges is critical to ensure the model's reliability and usefulness in real-world medical applications.

8. Conclusions and future work

In this study, the performance of six machine learning algorithms, including XGBoost, KNN, RF, LG, BRR, and SVM, were examined to predict the hospital disposition for trauma patients, mainly whether a patient will be safely discharged from the hospital or not. In addition, a univariate LG was used to acquire the set of features that provide highly accurate prediction outcomes. To evaluate the developed models, different evaluation metrics were utilized to assess the prediction accuracy. The evaluation metrics were accuracy, recall, P, F1 score, and AUC. By examining the outputs of the best forecasting model and the efficiency of the estimate algorithms, the XGBoost algorithm demonstrated a high degree of detection and prediction accuracy for disposed-to-home patients: of the 6059 patients that were sent home, XGBoost correctly predicted 5944 patients, or 98% of the total. Furthermore, the XGBoost algorithm accurately predicted 228 out of 274 hospital deaths, thus demonstrating its superior detection accuracy of deceased patients.

Finally, by utilizing the study's developed framework, medical teams would have an invaluable tool to help them decide on treatment plans and allot hospital resources. Moreover, insurance companies can utilize this data to project treatment costs and make appropriate plans. For future work and to enhance patient outcomes, our work emphasizes the potential of ML in healthcare, as well as the significance of creating trustworthy prediction models. Future studies can expand the dataset to include more varied patient populations and explore how this approach might be applied to other healthcare outcomes. In addition, this study investigated the performance of different machine learning algorithms, and other studies can further examine the performance of deep learning algorithms, such as long-term short memory and convolution neural networks.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Researchers would like to thank the Deanship of Scientific Research, Qassim University for funding publication of this project.

Conflict of interest

The authors declare that they have no competing interests.

References

1. M. K. Delgado, M. A. Yokell, K. L. Staudenmayer, D. A. Spain, T. Hernandez-Boussard, N. E. Wang, Factors associated with the disposition of severely injured patients initially seen at non-trauma center emergency departments: disparities by insurance status, *JAMA Surg.*, **149** (2014), 422–430. https://doi.org/10.1001/jamasurg.2013.4398
2. S. Y. Lee, R. B. Chinnam, E. Dalkiran, S. Krupp, M. Nauss, Prediction of emergency department patient disposition decision for proactive resource allocation for admission, *Healthcare Manag. Sci.*, **23** (2020), 339–359. https://doi.org/10.1007/s10729-019-09496-y
3. S. M. Fernando, B. Rochwerg, P. M. Reardon, K. Thavorn, A. J. Seely, J. J. Perry, et al., Emergency department disposition decisions and associated mortality and costs in ICU patients with suspected infection, *Crit. Care*, **22** (2018), 172. https://doi.org/10.1186/s13054-018-2096-8
4. A. G. Rapsang, D. C. Shyam, Scoring systems of severity in patients with multiple trauma, *Cir. Esp.*, **93** (2015), 213–221. https://doi.org/10.1016/j.ciresp.2013.12.021
5. J. Plummer, H. Brown, K. Jones, D. Fearon-Boothe, N. Meeks-Aitken, A. McDonald, Trauma: the burden of a preventable problem, *West Indian Med. J.*, **59** (2010), 26.
6. M. Graham, P. Parikh, S. Hirpara, M. C. McCarthy, E. R. Haut, P. P. Parikh, Predicting discharge disposition in trauma patients: development, validation, and generalization of a model using the national trauma data bank, *Am. Surg.*, **86** (2020), 1703–1709. https://doi.org/2010.1177/0003134820949523
7. D. Cuadrado, A. Valls, D. Riaño, Predicting intensive care unit patients' discharge date with a hybrid machine learning model that combines length of stay and days to discharge, *Mathematics*, **11** (2023), 4773. https://doi.org/10.3390/math11234773
8. B. Stocker, H. K. Weiss, N. Weingarten, K. E. Engelhardt, M. Engoren, J. Posluszny, Challenges in predicting discharge disposition for trauma and emergency general surgery patients, *J. Surg. Res.*, **256** (2021), 278–288. https://doi.org/10.1016/j.jss.2021.03.014
9. C. F. Mickle, D. Deb, Early prediction of patient discharge disposition in acute neurological care using machine learning, *BMC Health Serv. Res.*, **22** (2022), 1281. https://doi.org/10.1186/s12913-022-08615-w
10. M. A. Abd-Elrazek, A. A. Eltahawi, M. H. Abd Elaziz, M. N. Abd-Elwhab, Predicting length of stay in hospitals intensive care unit using general admission features, *Ain Shams Eng. J.*, **12** (2021), 3691–3702. https://doi.org/10.1016/j.asej.2021.02.018
11. H. Zhong, B. Wang, D. Wang, Z. Liu, C. Xing, Y. Wu, et al., The application of machine learning algorithms in predicting the length of stay following femoral neck fracture, *Int. J. Med. Inf.*, **155** (2021), 104572. https://doi.org/10.1016/j.ijmedinf.2021.104572

12. J. Liu, C. M. M. Lin, F. Chao, Gradient boost with convolution neural network for stock forecast, *Proceedings of the Advances in Computational Intelligence Systems: Contributions Presented at the 19th UK Workshop on Computational Intelligence*, Springer, 2020, 155–165. https://doi.org/10.1007/978-3-030-29933-0_13
13. J. Friedman, T. Hastie, R. Tibshirani, Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors), *Ann. Stat.*, **28** (2000), 337–340. https://doi.org/10.1214/aos/1016218223
14. M. Steinbach, P. N. Tan, kNN: k-nearest neighbors, *The Top Ten Algorithms in Data Mining*, 2009, 151–162.
15. S. Dhanabal, S. Chandramathi, A review of various k-nearest neighbor query processing techniques, *Int. J. Comput. Appl.*, **31** (2011), 14–22. https://doi.org/10.5120/3836-5332
16. L. Breiman, Random forests, *Mach. Learn.*, **45** (2001), 5–32. https://doi.org/10.1023/A:1010933404324
17. T. K. Ho, Random decision forests, *Proceedings of the 3rd International Conference on Document Analysis and Recognition*, 1995, 278–282. https://doi.org/10.1109/ICDAR.1995.598994
18. A. Cutler, D. R. Cutler, J. R. Stevens, Random forests, In: C. Zhang, Y. Ma, *Ensemble machine learning*, Springer, 2012, 157–175. http://doi.org/10.1007/978-1-4419-9326-7_5
19. Ö. Çokluk, Logistic regression: concept and application, *Educ. Sci. Theory Pract.*, **10** (2010), 1397–1407.
20. A. Y. Ng, M. I. Jordan, On discriminative vs. generative classifiers: a comparison of logistic regression and naive bayes, *Adv. Neural Inf. Process. Syst.*, **14** (2001), 841–848. https://doi.org/10.5555/2980539.2980648
21. G. C. McDonald, Ridge regression, *Wiley Interdiscip. Rev. Comput. Stat.*, **1** (2009), 93–100. https://doi.org/10.1002/wics.14
22. C. M. Bishop, M. E. Tipping, Bayesian regression and classification, *Adv. Learn. Theory*, **190** (2003), 267–288.
23. C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.*, **20** (1995), 273–297. https://doi.org/10.1007/BF00994018
24. M. Awad, R. Khanna, *Support vector machines for classification*, Springer, 2015, 39–66. https://doi.org/10.1007/978-1-4302-5990-9_3
25. S. Ghosh, A. Dasgupta, A. Swetapadma, A study on support vector machine-based linear and non-linear pattern classification, *2019 International Conference on Intelligent Sustainable Systems (ICISS)*, 2019, 24–28. https://doi.org/10.1109/ISS1.2019.8908018
26. A. Mansoori, M. Zeinalnezhad, L. Nazarimanesh, Optimization of tree-based machine learning models to predict the length of hospital stay using genetic algorithm, *J. Healthcare Eng.*, **2023** (2023), 9673395. https://doi.org/10.1155/2023/9673395
27. M. M. Alam, An efficient random forest algorithm-based telemonitoring framework to predict mortality and length of stay of patients in ICU, *Multimedia Tools Appl.*, 2023. https://doi.org/10.1007/s11042-023-17239-z
28. B. Eftekhar, K. Mohammad, H. E. Ardebili, M. Ghodsi, E. Ketabchi, Comparison of artificial neural network and logistic regression models for prediction of mortality in head trauma based on initial clinical data. *BMC Med. Inf. Decis. Mak.*, **5** (2005), 3. https://doi.org/10.1186/1472-6947-5-3

29. K. Li, H. Wu, F. Pan, L. Chen, C. Feng, Y. Liu, et al., A machine learning-based model to predict acute traumatic coagulopathy in trauma patients upon emergency hospitalization, *Clin. Appl. Thromb. Hemost.*, **26** (2020), 1076029619897827. https://doi.org/10.1177/1076029619897827
30. S. Lee, W. S. Kang, S. Seo, D. W. Kim, H. Ko, J. Kim, et al., Model for predicting in-hospital mortality of physical trauma patients using artificial intelligence techniques: nationwide population-based study in Korea, *J. Med. Int. Res.*, **24** (2022), e43757. https://doi.org/10.2196/43757
31. S. D. Hsu, E. Chao, S. J. Chen, D. Y. Hueng, H. Y. Lan, H. H. Chiang, Machine learning algorithms to predict in-hospital mortality in patients with traumatic brain injury, *J. Pers. Med.*, **11** (2021), 1144. https://doi.org/10.3390/jpm11111144
32. R. Wang, L. Wang, J. Zhang, M. He, J. Xu, XGBoost machine learning algorism performed better than regression models in predicting mortality of moderate-to-severe traumatic brain injury, *World Neurosurg.*, **163** (2022), 617–622. https://doi.org/10.1016/j.wneu.2022.04.044
33. J. S. Kong, K. H. Lee, O. H. Kim, H. Y. Lee, C. Y. Kang, D. Choi, et al., Machine learning-based injury severity prediction of level 1 trauma center enrolled patients associated with car-to-car crashes in Korea, *Comput. Biol. Med.*, **153** (2023), 106393. https://doi.org/10.1016/j.compbiomed.2022.106393
34. C. S. Rau, S. C. Wu, J. F. Chuang, C. Y. Huang, H. T. Liu, P. C. Chien, et al., Machine learning models of survival prediction in trauma patients, *J. Clin. Med.*, **8** (2019), 799. https://doi.org/10.3390/jcm8060799
35. I. H. Witten, E. Frank, M. A. Hall, C. J. Pal, Appendix B-the WEKA workbench, In: *Data mining*, 4 Eds., Morgan Kaufmann, 2017, 553–571. https://doi.org/10.1016/B978-0-12-804291-5.00024-6
36. W. S. Hong, A. D. Haimovich, R. A. Taylor, Predicting hospital admission at emergency department triage using machine learning, *PloS One*, **13** (2018), 0201016. https://doi.org/10.1371/journal.pone.0201016
37. A. K. Zalikha, T. Court, F. Nham, M. M. El-Othmani, R. P. Shah, Improved performance of machine learning models in predicting length of stay, discharge disposition, and inpatient mortality after total knee arthroplasty using patient-specific variables, *Arthroplasty*, **5** (2023), 31. https://doi.org/10.1186/s42836-023-00187-2