SAT 5141 Clinical Decision Support Systems

Mortality Prediction for In-Hospital ICU Patients

Rishidhar Konathala , Chandu Narasimhareddyvari

Course Project Group 8

Professor: Dr.Guy Hembroff

Michigan Tech College of Applied Computing

03rd December 2024

Introduction

Objective

- **Goal:** Develop machine learning models to predict in-hospital mortality for ICU patients
- **Data:** First 48 hours of ICU admission from **MIMIC II database**
- **Dataset Link:** PhysioNet Challenge 2012
- **Proposed models :** XG Boost and ANN
- **Evaluation :**Evaluate the model's performance using key metrics like **AUC-ROC**, **accuracy**, and **F1-score**.

Challenges

- **Missing Data:** Frequent in ICU datasets due to skipped measurements.
- **Time-Series Processing:** ICU metrics often captured as time-series data.
- **Class Imbalance:** Minority class (mortality cases) underrepresented.

Background of the problem

- **ICU mortality prediction** is vital for assessing treatment efficacy, resource allocation, and patient outcomes.

Each year (in the USA)

- Over **5 million I**CU patients are treated annually in the USA, with mortality rates of **8–19%.**
- Accurate predictions aid clinicians in care planning and resource management.
- Early risk identification reduces complications, optimizes interventions, and lowers costs.

ICU mortality prediction

Traditional scoring systems like **APACHE, SAPS, and SOFA** rely on manually designed features but have limitations in scalability and accuracy.(https://clincalc.com/IcuMortality/)

A Neural Network Model for Mortality Prediction in ICU

Henian Xia[1], Brian J Daley[2], Adam Petrie[3], Xiaopeng Zhao[1]

[1]Department of Mechanical, Aerospace and Biomedical Engineering, University of Tennessee, Knoxville, USA
[2]Department of Surgery, University of Tennessee Medical Center, Knoxville, USA
[3]Department of Statistics, Operations, and Management Science, University of Tennessee, Knoxville, USA

Abstract

Scoring the severity of illness of ICU patients can provide evaluation of a patient's situation and thus help doctors make decisions on what treatment to take. This study aimed to develop an artificial neural network model for patient-specific prediction of in-hospital mortality. Data from PhysioNet Challenge 2012 was used. 12,000 records were divided to a training set, a test set and a validation set, each of which contains 4000 records. Outcomes are provided for the training set. A neural network model was developed to predict the risk of in-hospital mortality using various physiological measurements from the ICU. Twenty-six features were selected after a thorough investigation over the different variables and features. A two-layer neural network with fifteen neurons in the hidden layer was used for classification. One hundred voting classifiers were trained and the model's output was the average of the one hundred outputs. A fuzzy threshold was utilized to determine the outcome of each record from the output of the network. Our model yielded an event 1 score of 0.5088 and an event 2 score of 82.211 on the test data set.

1. Introduction

An intensive care unit (ICU) is for patients with the most serious diseases or injuries. Most of the patients need support from equipment like the medical ventilator to maintain normal body functions and need to be constantly and closely monitored. For decades, the number of ICUs has experienced a worldwide increase [1]. During the ICU stay, different physiological parameters are measured and analysed each day. Those parameters are used in scoring systems to gauge the severity of the patients. Many types of severity or prognostic scoring systems have been developed for the ICU, such as the acute physiology and chronic health evaluation system (APACHE II), the simplified acute physiology score (SAPS II) and the mortality probability model (MPM). Those systems are important for many reasons. They provide evaluation of patients' situations so that the intensive care can be restricted to patients most at need. While the intensive care improves the outcome for seriously ill patients, it comes with an expensive cost. In 2005, the mean intensive care unit cost is as high as 31,574±42,570 dollars for patients requiring mechanical ventilation and 12,931 ± 20,569 dollars for those not requiring mechanical ventilation [2]. The mortality assessment is crucial for making the critical decision of whether to interrupt the life-support treatments when intensive care is considered helpless. Besides, the mortality prediction helps doctors determine what treatment process to take.

Most of the prevalent mortality assessment models are developed using linear regression over a score computed from physiological variables. For example, the SAPS II examined 37 variables, and chose 17 that were found to be associated with the hospital mortality most significantly. The 17 variables include 12 physiology variables, age, type of admission (scheduled surgical, unscheduled surgical, or medical), and three underlying disease variables (acquired immunodeficiency syndrome, metastatic cancer, and hematologic malignancy). A score is computed using the 17 variables and is converted to a probability of hospital mortality using a linear regression equation.

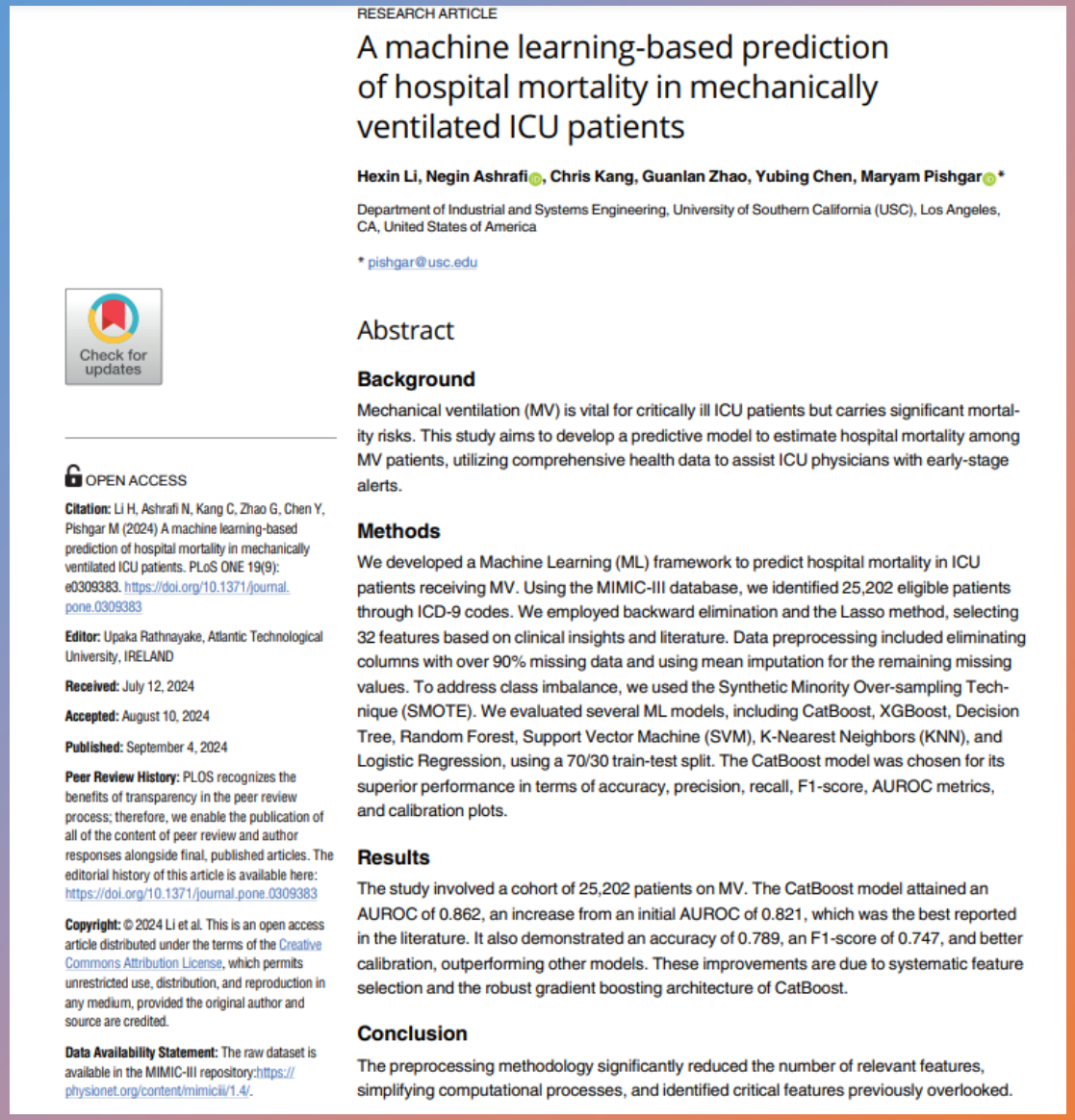
More recently, the data mining techniques have been proved to be useful in the ICU mortality prediction [3, 4]. The data mining techniques are used to discover patterns hidden in large clinical data [5]. The volume of clinical data is increasing every single day. It is difficult for human experts to extract information from the data by looking at them manually. In contrast, the data mining techniques can automatically extract information from the raw data [6].

Literature Review

- Study developed ANN model to predict in-hospital mortality for ICU patients
- Used data from PhysioNet Challenge 2012 (12,000 records)
- Selected 26 features from physiological measurements

- **Model architecture:**
  - Two-layer neural network
  - 15 neurons in the hidden layer
  - 100 voting classifiers used
- **Techniques employed:**
  - Oversampling of positive (mortality) cases
  - Fuzzy threshold for classification
- **Performance:**
  - Event 1 score: 0.5088 on test data
  - Event 2 score: 82.211 on test data
- **Key features:** GCS, HCO3, BUN, urine output, vital signs

- **Ref :**https://www.cinc.org/archives/2012/pdf/0261.pdf



RESEARCH ARTICLE

A machine learning-based prediction of hospital mortality in mechanically ventilated ICU patients

Hexin Li, Negin Ashrafi, Chris Kang, Guanlan Zhao, Yubing Chen, Maryam Pishgar*

Department of Industrial and Systems Engineering, University of Southern California (USC), Los Angeles, CA, United States of America

* pishgar@usc.edu

Abstract

Background

Mechanical ventilation (MV) is vital for critically ill ICU patients but carries significant mortality risks. This study aims to develop a predictive model to estimate hospital mortality among MV patients, utilizing comprehensive health data to assist ICU physicians with early-stage alerts.

Methods

We developed a Machine Learning (ML) framework to predict hospital mortality in ICU patients receiving MV. Using the MIMIC-III database, we identified 25,202 eligible patients through ICD-9 codes. We employed backward elimination and the Lasso method, selecting 32 features based on clinical insights and literature. Data preprocessing included eliminating columns with over 90% missing data and using mean imputation for the remaining missing values. To address class imbalance, we used the Synthetic Minority Over-sampling Technique (SMOTE). We evaluated several ML models, including CatBoost, XGBoost, Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Logistic Regression, using a 70/30 train-test split. The CatBoost model was chosen for its superior performance in terms of accuracy, precision, recall, F1-score, AUROC metrics, and calibration plots.

Results

The study involved a cohort of 25,202 patients on MV. The CatBoost model attained an AUROC of 0.862, an increase from an initial AUROC of 0.821, which was the best reported in the literature. It also demonstrated an accuracy of 0.789, an F1-score of 0.747, and better calibration, outperforming other models. These improvements are due to systematic feature selection and the robust gradient boosting architecture of CatBoost.

Conclusion

The preprocessing methodology significantly reduced the number of relevant features, simplifying computational processes, and identified critical features previously overlooked.

Check for updates

OPEN ACCESS

Citation: Li H, Ashrafi N, Kang C, Zhao G, Chen Y, Pishgar M (2024) A machine learning-based prediction of hospital mortality in mechanically ventilated ICU patients. PLoS ONE 19(9): e0309383. https://doi.org/10.1371/journal.pone.0309383

Editor: Upaka Rathnayake, Atlantic Technological University, IRELAND

Received: July 12, 2024

Accepted: August 10, 2024

Published: September 4, 2024

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pone.0309383

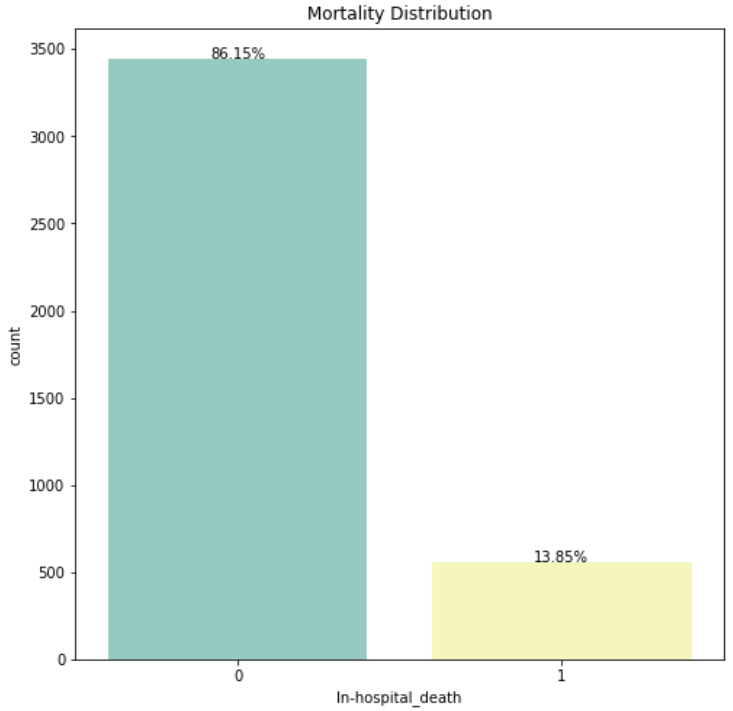
Data Availability Statement: The raw dataset is available in the MIMIC-III repository:https://physionet.org/content/mimiciii/1.4/.

Literature Review

- Study developed ML model to predict mortality in ICU patients on mechanical ventilation
- • Used MIMIC-III database with 25,202 eligible patients
- • Selected 32 features using backward elimination, Lasso method, and clinical insights
- • Addressed class imbalance using SMOTE technique
- • Evaluated multiple ML models: CatBoost, XGBoost, Decision Tree, Random Forest, SVM, KNN, Logistic Regression
- **CatBoost model performed best:**
- AUROC: 0.862
- Accuracy: 0.789
- F1-score: 0.747
- • Improved upon the previous best-reported AUROC of 0.821
- • Model shows potential to enhance resource allocation and personalized interventions in ICUs
-
- **Ref:https://pubmed.ncbi.nlm.nih.gov/39231126/**

Dataset

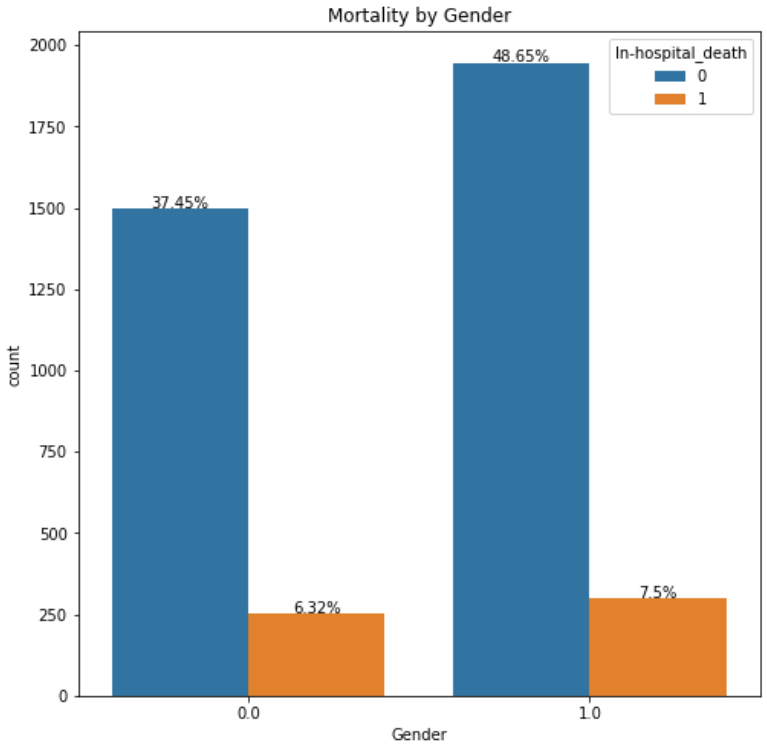
- **Size:** 12,000+ ICU admissions
- **Time Frame:** First 48 hours of ICU stay
- **Data Types:**
  - **Demographic:** age, gender, height, ICU type, and initial weight
  - **Physiological:** vital signs, lab metrics (time series)
- **Outcome:** Binary in-hospital death (target class (`In-hospital_death`))
- **Key Features:**
  - 5 demographic descriptors
  - 36 physiological time series
  - APACHE, SAPS, and SOFA
- **Class Distribution:** 13.85 % mortality rate

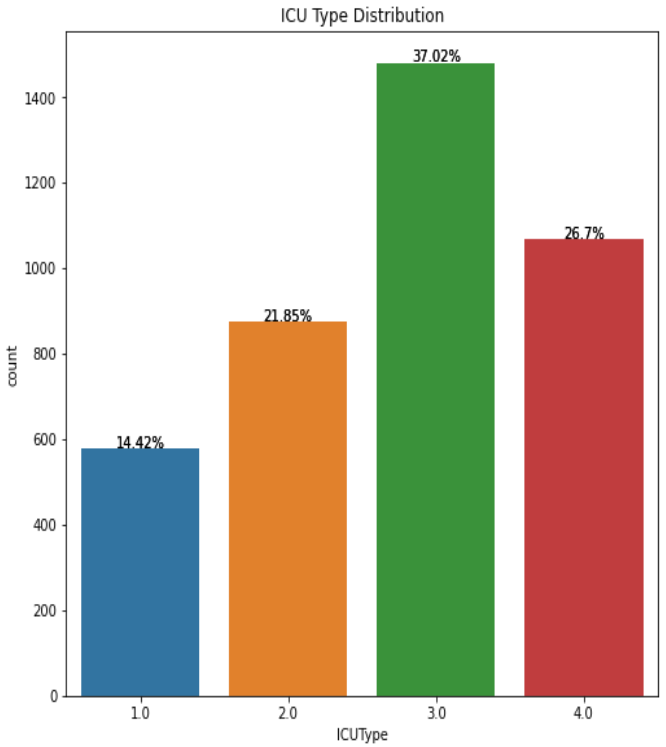


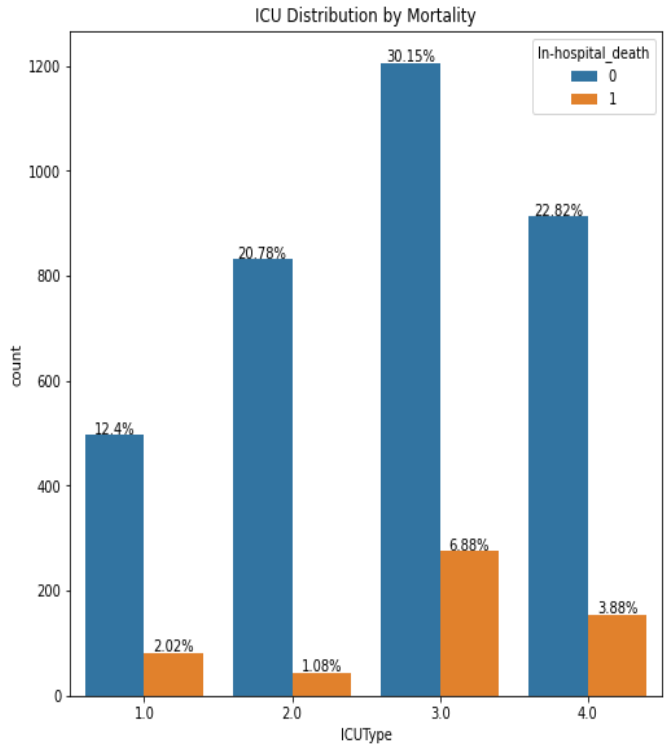
	RecordID	Age	Gender	Height	Weight	Urine	HR	Temp	NIDiasABP	SysABP	...	pH	PaCO2	PaO2	Platelets
0	137021.0	66.0	1.0	177.8	102.00	117.50	113.41	37.23	56.61	NaN	...	NaN	NaN	NaN	209.00
0	133611.0	50.0	1.0	NaN	92.40	210.51	85.93	38.04	58.60	126.69	...	7.40	41.00	175.00	243.33
0	137860.0	40.0	1.0	185.4	81.80	467.41	98.83	36.83	57.88	157.63	...	7.36	38.82	127.12	143.33
0	134781.0	58.0	1.0	180.3	99.22	86.55	81.63	36.96	49.64	113.86	...	7.36	46.20	147.60	109.60
0	133534.0	80.0	1.0	NaN	73.00	194.12	71.86	37.10	72.85	NaN	...	NaN	NaN	NaN	NaN

Data Distribution

Visualizing Data Distributions







Preprocessing

Handling Missing Values

- Imputed missing physiological and lab values using **mean imputation**.
- Median imputation for **height** and **weight** due to skewed distributions.

Mortality Distribution

- Analyzed ICU mortality distribution:
  - **Observation:** Mortality rates were consistent across genders but varied significantly by age and BMI.
  - ICU Type 3 and 4 patients exhibited higher mortality.

Age Distribution

- ICU patients skew older; **higher mortality rates** observed in age groups **above 60 years.**

BMI Analysis

- **Trends:**
  - Higher mortality rates among patients with BMI categorized as **Underweight** or **Obese**.
  - Gender disparity observed, with more males admitted than females.

Preprocessing (cont..)

- **Standardization**
  - Scaled continuous variables to ensure consistent scaling across features.
- **Encoding**
  - Categorical variables (e.g., **ICU type**, **gender**) encoded using **label encoding**.
- **Class Imbalance :**
  - Class Imbalance is handled to ensure that the model doesn’t favor the **majority class**, which is critical for healthcare applications where identifying rare outcomes like mortality is crucial.
  - Addressed using techniques like **SMOTE (Synthetic Minority Over-sampling Technique).**
  - and **oversampling**.

Model Development

Artificial Neural Network (ANN)

Architecture:

- A feed-forward neural network designed for binary classification tasks.
- Captures complex, non-linear relationships in ICU patient data.

Layers:

- **Input Layer:** Takes preprocessed features (22 features from ICU data) as input.
- **Hidden Layers:**
  - Dense Layer 1: 20 units, ReLU activation.
  - Dense Layer 2: 40 units, ReLU activation.
  - Dense Layer 3: 80 units, ReLU activation.
  - Dense Layer 4: 40 units, ReLU activation.
  - Dense Layer 5: 20 units, ReLU activation.
- **Output Layer:**
  - Dense Layer 6: 1unit, Sigmoid activation (outputs mortality probability).

Model Development

XGBoost Classifier

- **Input Layer:**
  - Takes preprocessed features (e.g., ICU data with 22 features) as input.
- **Base Learner:**
  - Decision Trees:
    - Trees are built sequentially, with each tree focusing on correcting errors (residuals) from previous trees.
    - Maximum tree depth: **5** (controls complexity).
- **Ensemble Learning:**
  - Combines predictions from **100 decision trees** using gradient boosting.
  - Optimized to minimize the **binary logistic loss** function.
- **Regularization:**
  - L1 and L2 regularization to prevent overfitting.
  - Learning rate: **0.1** to control the contribution of each tree.
- **Output Layer:**
  - Produces the final prediction by summing weighted outputs of all trees.
  - Applies a logistic transformation for binary classification (In-hospital_death probability).

Model Training

ANN Model :

Epochs: 100

Batch Size: 32

Learning Rate: (default Adam learning rate)

Optimizer: Adam

Loss Function: Binary Cross-Entropy (BCE)

Metrics: Accuracy

Class Weights: Optionally applied (if weights ≠ -1)

XG Boost :

n_estimators: 100 decision trees used for boosting.

Learning Rate: 0.1 (controls the contribution of each tree).

Max Depth: 5 (limits tree depth to prevent overfitting).

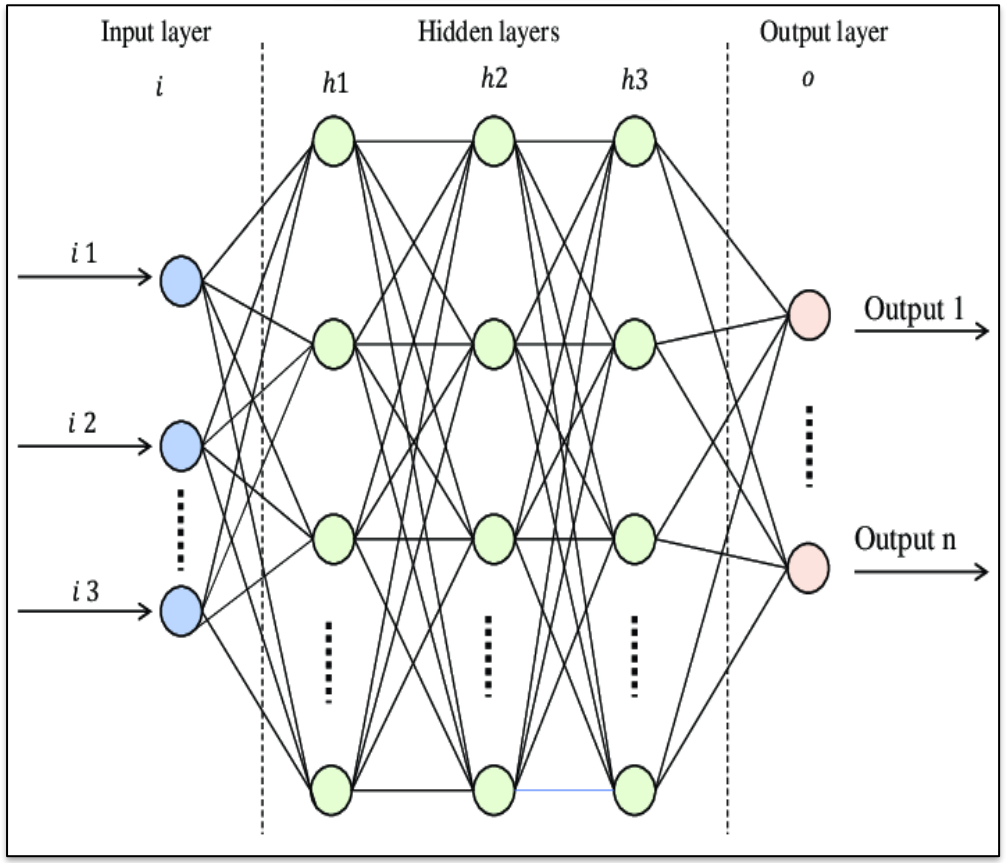
Random State: 42

Objective: Binary logistic regression for mortality classification.

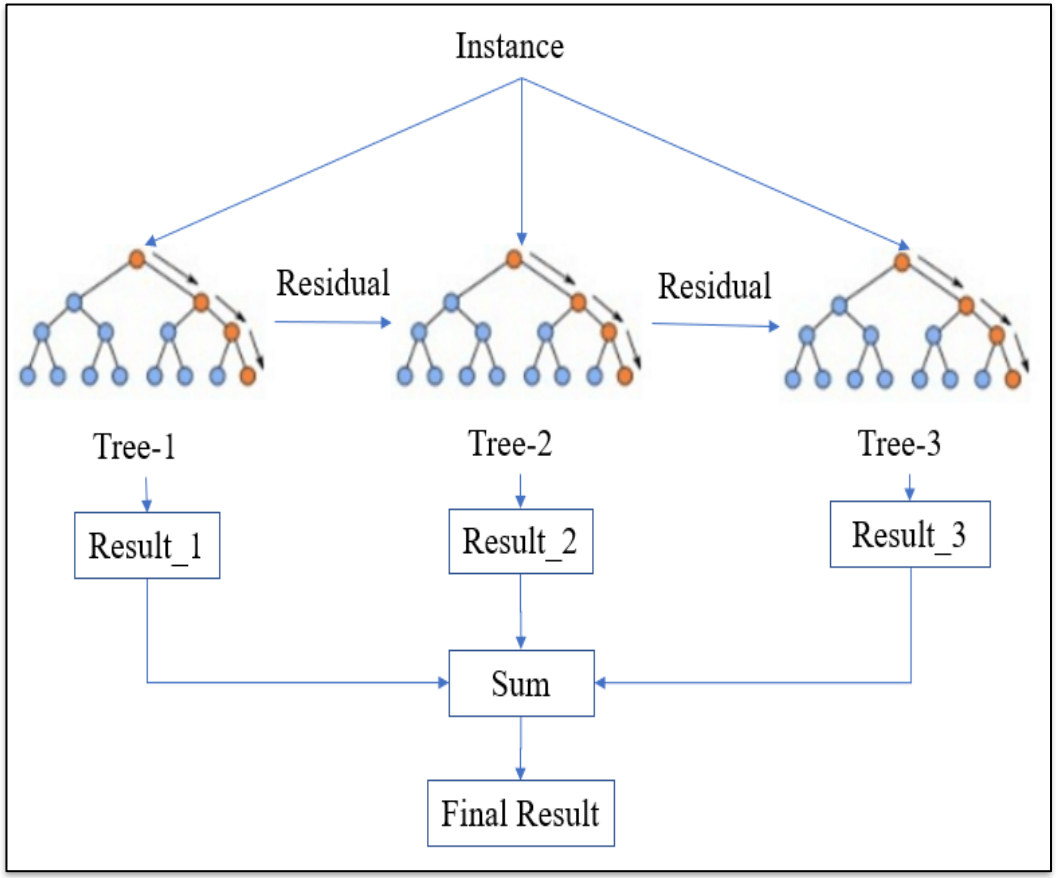
Feature Importance: Extracts and ranks critical features influencing predictions.

Architecture Overview

FNN



XG Boost



Performance Metrics： Artificial Neural Networks (ANN)

```
Training with Oversampled Dataset:
Epoch 1/100
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarning:

Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential mode

173/173 ━━━━━━━━━━━━━━━━━━━━ 2s 2ms/step - accuracy: 0.6193 - loss: 0.6501
Epoch 2/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.7497 - loss: 0.5123
Epoch 3/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.7780 - loss: 0.4612
Epoch 4/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.7984 - loss: 0.4271
Epoch 5/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8163 - loss: 0.3971
Epoch 6/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8332 - loss: 0.3706
Epoch 7/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8511 - loss: 0.3521
Epoch 8/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8610 - loss: 0.3305
Epoch 9/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8611 - loss: 0.3150
Epoch 10/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8798 - loss: 0.2980
Epoch 11/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8882 - loss: 0.2802
Epoch 12/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8970 - loss: 0.2620
Epoch 13/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.9054 - loss: 0.2425
...
    accuracy                           0.93      1379
   macro avg       0.94      0.93      0.93      1379
weighted avg       0.94      0.93      0.93      1379
```

```
Training with SMOTE Dataset:
Epoch 1/100
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarning:

Do not pass an `input_shape`/`input_dim` argument to a layer. When using Sequential mode

173/173 ━━━━━━━━━━━━━━━━━━━━ 3s 5ms/step - accuracy: 0.6310 - loss: 0.6400
Epoch 2/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.7485 - loss: 0.5199
Epoch 3/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.7776 - loss: 0.4750
Epoch 4/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.7957 - loss: 0.4438
Epoch 5/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8130 - loss: 0.4216
Epoch 6/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8224 - loss: 0.4019
Epoch 7/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8348 - loss: 0.3845
Epoch 8/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8428 - loss: 0.3670
Epoch 9/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8570 - loss: 0.3488
Epoch 10/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8677 - loss: 0.3295
Epoch 11/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8742 - loss: 0.3152
Epoch 12/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8811 - loss: 0.3007
Epoch 13/100
173/173 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8878 - loss: 0.2872
...
    accuracy                           0.89      1379
   macro avg       0.90      0.89      0.89      1379
weighted avg       0.90      0.89      0.89      1379
```

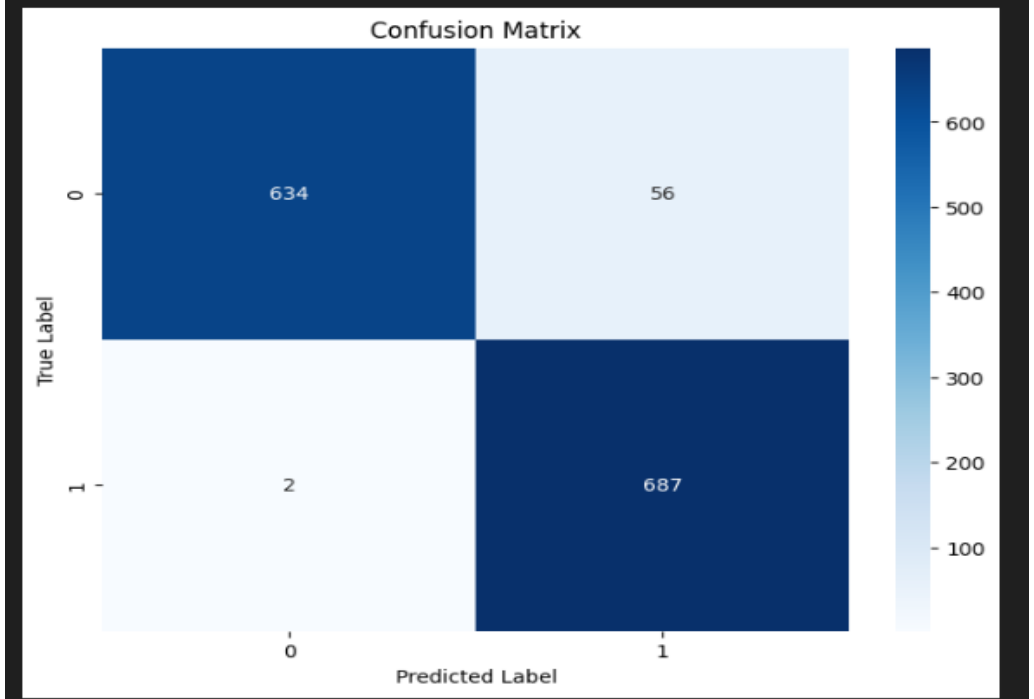
XGboost classifier:

XGboost on oversampled data

XGBoost Classifier Results:
Accuracy: 0.9579405366207396

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.92	0.96	690
1	0.92	1.00	0.96	689
accuracy			0.96	1379
macro avg	0.96	0.96	0.96	1379
weighted avg	0.96	0.96	0.96	1379

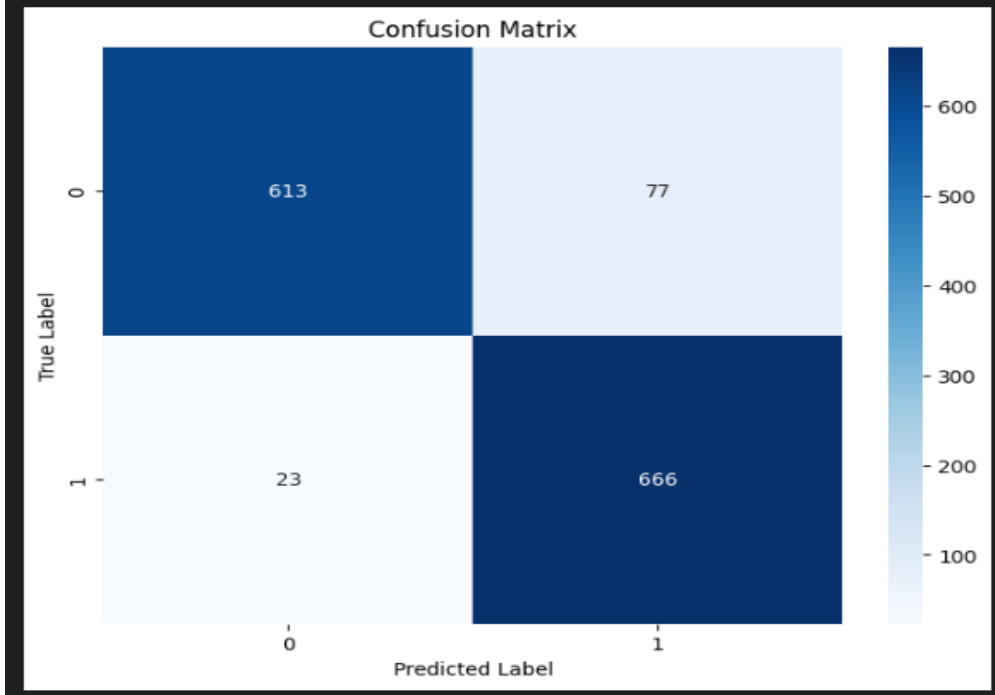


XGboost on Smote data

XGBoost Classifier Results:
Accuracy: 0.9274836838288615

Classification Report:

	precision	recall	f1-score	support
0	0.96	0.89	0.92	690
1	0.90	0.97	0.93	689
accuracy			0.93	1379
macro avg	0.93	0.93	0.93	1379
weighted avg	0.93	0.93	0.93	1379



Results

XGBoost classifier

- **Accuracy:**
  Oversampled: 0.95
  Smote: 0.93
- **Precision:**
  Oversampled: 1
  Smote: 0.92
- **Recall:**
  Oversampled: 0.92
  Smote: 0.89
- **F-1 score:**
  Oversampled: 0.96
  Smote: 0.96

ANN

- **Accuracy:**
  Oversampled: 0.93
  Smote: 0.89
- **Precision:**
  Oversampled: 0.88
  Smote: 0.85
- **Recall:**
  Oversampled: 0.86
  Smote: 0.95
- **F-1 score:**
  Oversampled: 0.92
  Smote: 0.90

Limitations and Future work

Limitations:

Dataset challenges:

- Smaller dataset which may restrict generalizability.
- Handling of missing values might lead to bias.

Class imbalance and model interpretability:

- Despite using SMOTE technique, minority class prediction might still be suboptimal.
- ANN and XGBoost lack intuitive interpretability, making clinical application challenging.

Future work

- **Enhance Interpretability**:
  - Implement interpretable models like decision trees or logistic regression for explainable predictions alongside ANN and XGBoost.
  - Using attention mechanisms in neural networks to highlight critical features.
- **Dynamic Data Utilization**:
  - Expansion of the scope to include real-time ICU data streams to predict mortality trends dynamically.
  - Integration of continuous patient monitoring data for richer insights.
- **Bias Mitigation**:
  - Investigate potential biases in model predictions, such as demographic or socio-economic disparities.
  - Incorporate fairness metrics to evaluate the impact of the models on diverse patient groups.
- **Incorporate Additional Predictors**:
  - Consider non-physiological factors like treatment regimens, ICU staffing, and patient histories to improve model accuracy.
  - Include genomics or biomarkers if available for personalized risk assessment.

Conclusion

- Upon comparing the models, XGBoost demonstrated superior precision and interpretability, making it ideal for scenarios requiring faster inference and clear, explainable results. Conversely, ANN outperformed in recall and AUC-ROC, making it more suitable for healthcare applications prioritizing sensitivity to mortality risk, particularly in critical care.

- Integrating XGBoost and ANN within an ensemble framework could harness their complementary strengths. Continued research on enhancing explainability and seamless integration with real-time ICU systems is crucial for developing actionable clinical decision-support tools.

- In summary, while both models performed well, XGBoost exhibited better overall performance on the given dataset.

Thank You