ICU Patient Report

By: Taylen Inthisane

Introduction

After conducting exploratory data analysis, the next critical steps in any data science project are data cleaning and selecting the most appropriate modeling techniques to achieve the desired analytical goals. This report outlines the preparation of the MIMIC-IV dataset prior to modeling, including variable selection, data cleaning, feature engineering, and final preprocessing steps. It then details the rationale behind the chosen models and how they align with the technical objective: identifying variables with a p-value less than 0.05 at a 95% confidence level. By leveraging this analysis, the goal is to generate actionable insights for the ICU department that can help reduce patient length of stay and mortality rates while preserving, and potentially enhancing, the quality of care delivered.

Section 1: Data Preparation

Section 1.1:Data Selection Rationale

Following a systematic evaluation, 30 of the original 132 variables were retained for analysis. The excluded variables fell into three primary categories:

1. **Time-Based Variables**: Exploratory data analysis (EDA) revealed that time-related features had minimal impact on patient outcomes. As a result, these variables were removed to streamline the dataset.
2. **Granularity Misalignment**: The analysis focuses on individual ICU visits and emphasizes variables that the ICU department can directly influence. Therefore, features such as Caregiver_ID and Order_Provider_ID were excluded, as they pertain to individual staff members rather than departmental practices. Similarly, detailed event-level data—such as microbiology events, pharmacy transactions, and POE records—were removed for being too granular and outside the scope of actionable departmental insights.
3. **Redundant Information**: Variables like ICU admission and discharge times were excluded because a derived length_of_stay (LOS) variable was already provided. Likewise, date_of_death

(DOD) was dropped in favor of the more analysis-friendly death_flag binary indicator (1 = deceased, 0 = survived).

This refined feature set ensures the analysis remains focused, interpretable, and aligned with the goal of identifying actionable insights to improve patient care within the ICU.

For this analysis, I've chosen to use fourteen years of data from the MIMIC-IV dataset. This timeframe offers a robust and comprehensive sample that has not been previously analyzed, making it ideal for uncovering meaningful insights. My focus is less on the specific dates and more on the clinical procedures that influence patient outcomes particularly length of stay (LOS) and mortality. While seasonal factors, such as increased flu cases during winter months, do play a role, incorporating timestamp-level data introduces excessive granularity and unnecessary complexity. By concentrating on broader clinical patterns rather than minute-by-minute variations, the analysis remains both manageable and clinically relevant.

One important consideration in this analysis is the potential obsolescence of clinical procedures over time. To account for this, I examined the ICD version codes ICD-9 representing older procedures and ICD-10 reflecting more recent updates. While reviewing the data, I considered whether to exclude variables based on their ICD version. However, I ultimately chose not to differentiate between them. My goal is to uncover trends across all procedural data, regardless of version. In fact, some newer codes may correspond to less effective interventions, and I wanted to ensure those possibilities were captured in the analysis.

To maintain a clear and focused scope for this project, the data selection process prioritized removing information outside the ICU department's direct control. This meant excluding data from sources such as pharmacy records, lab events, and microbiology reports. Instead, I concentrated on variables the ICU team can act upon such as diagnoses, clinical procedures, and prescriptions which offer valuable context for understanding extended patient stays or mortality outcomes. When paired with patient demographics, this data helps reveal meaningful trends. After all, while we may be legally treated the same, our genetic differences can significantly influence clinical outcomes. For example, individuals with red hair often require higher doses of anesthesia due to mutations in their melanocortin-1 receptors, which increase their resistance. If these genetic factors aren't considered, such patients may experience more pain than others under standard dosing protocols.

Section 1.2: Data Cleaning Process

During data preprocessing, one of the key steps I focused on was identifying and addressing missing values. Variables with a high percentage of missing data warranted closer examination. For instance, while I initially considered including the Omar dataset, I found that certain fields such as Standing Blood Pressure (1 Min) had over 99% missing values. Given the lack of usable data, I chose to exclude those variables from the analysis. On the other hand, handling missing data for the "language" variable was more straightforward. Since it was a categorical field with only two possible values English or Unknown I assigned any missing entries to the "Unknown" category, ensuring consistency without compromising the dataset's integrity.

Since this analysis focuses on identifying which procedures and demographic factors influence patient length of stay and mortality, the concept of outliers is less rigid. While most data points fell within expected ranges, one patient had an exceptionally long stay. Rather than excluding this case, I chose to retain it in the dataset. My reasoning is rooted in representation. Patients face a wide range of circumstances and the removal of such data risks overlooking those on the margins. Although this may be a single instance in the dataset, it could reflect hundreds or even thousands of similar cases across the broader metropolitan area. Preserving this outlier ensures the analysis remains inclusive and reflective of real-world variability.

To address duplicate entries, I performed a thorough check for repeated records. Specifically, if a patient, for example patient 123, was diagnosed with a fever during their first visit, that record was retained only once. Any exact duplicates were removed to preserve the integrity of the data while still allowing for multiple visits per patient. Additionally, I standardized the dataset by stripping whitespace and converting all variable names to lowercase, ensuring consistency across fields.

Overall, data cleaning had minimal impact on the dataset, as most of the variables I intended to use were already complete. However, a challenge emerged during the data merging process due to one-to-many relationships—such as a single patient having multiple diagnoses. This caused the dataset to expand significantly, complicating the merge. I resolved this issue through aggregation, which is discussed in more detail in Section 1.4.

Section 1.3 Feature Engineering

Based on the data, I decided to derive the following variables.

- is_elderly : Variable is derived from anchor_age and the patient must have an anchor_age greater than 65 years old.
    - Elderly patients tend to have weaker immune systems. By isolating them and creating a category just for elderly patients we can derive age trends.
- los_hours: Derived from the length of stay multiplied by 24 hours. This is because describing los in 2.2 days is not as precise as los_hours.
    - Length of stay by hours is an easier variable to understand then los. This is because length of stay hours is smaller so the data can be more precise.
- long_stay_flag: Derived from length of stay. If the length of stay was greater than 7 days, then the length of stay gets assigned a long_stay_flag
    - If a patient's stay is longer than normal this variable allows us to view abnormal patient stays and can be used to help determine what patients needed more care/recovery.
- death_within_24h_flag= Derived from length of stay in hours (los_hours) which is less than 24 and where the death_flag equals 1.
    - Which patient died within being in the hospital for less than one day. Can use this to view trends in their data and potentially use this to put higher importance in future patients that come into the ICU with comparable diagnoses.
- has_diabetes: derived from searching if the patients diagnoses names are equal to diabetes
    - Diabetes is a common disease that has many medical implications. Knowing which patients has diabetes is important as it can potentially be a confounding variable which could affect LOS and death_flag
- has_cardiac_issues: derived from searching diagnoses names by if the diagnosis contains 'heart', 'cardiac', 'coronary'
    - Heart issues can have medical implications. Typically leads to longer recovery and can have longer mortality.
- num_diagnoses: derived by counting the number of diagnoses instances per patient
    - Created to see if the number of diagnoses a patient receives affects their LOS and death_flag
- num_procedures: derived by counting the number of procedure instances per patient
    - Created to see if the number of procedures a patient receives affects their LOS and death_flag
- num_drugs: derived by counting the number of drug instances per patient
    - Created to see if the number of drugs a patient receives affects their LOS and death_flag

- readmission_flag: by grouping the number of subject_id by stay_id. If the stay_id is greater than 1 then the patient was readmitted
    - If a patient is readmitted to the ICU, it can mean that either the ICU unit didn't do a good job diagnosing the issue or did a good job with a procedure or it can mean a patient is very sick.
- icu_type_encoded = derived by label encoding and fitting and transforming first_careunit.
    - Created for analysis. It is much easier to use number encoded numbers than using the entire care unit name
- Mortality_rate= derived by grouping the first_careunit and death flag and then getting the mean of data per care unit
    - Creating the mortality rate per care unit
- Mortality_rate_by_unit: Mapping first_careunit by mortality rate
    - Knowing the mortality rate by care unit can help identify which unit needs additional train and which care unit has a higher LOS and death_flag
- age_cap: Anything over 89 is represented as 90 (capping the data for HIPAA regulations)
    - Capping the age to follow HIPAA requirements
- age_group: grouping age capped anchor data and creating bins by 21, 40, 65, 80, 90
    - creating groups by age group. Some age groups may not have a long los or may not die as frequently compared to others
- log_los= Log transformation on length of stay to reduce right skew.
    - Transform the los data so it represents a more normal dataset.

Section 1.4 Data Integration and Final Preparation

To merge multiple data sources into a unified dataset, I used left joins based on the subject_id field. My initial goal was to maintain a wide format, allowing me to examine how hospital-issued care—specifically diagnoses, prescriptions, and procedures—impacted length of stay (LOS) and mortality. However, this approach quickly ran into performance issues. The resulting table became so large that Python couldn't handle the join due to memory limitations. The problem stemmed from differences in data granularity: for example, each row in the diagnosis table represented a single diagnosis per patient, whereas I needed a format where each patient was represented by a single row. To resolve this, I aggregated the data—combining multiple diagnoses into a single column rather than creating separate columns for each one. This preserved the necessary detail while keeping the dataset manageable and aligned with the analytical goals.

I formatted the data into a wide format, emphasizing that each entry corresponds to a patient's stay in the ICU.

Numeric columns: los, anchor_age, num_diagnoses, num_procedures, num_drugs, mortality_rate_by_unit, los_ hours, log_los

Categorical variables: subject_id, stay_id, first_careunit, gender, anchor_yeargroup, death_flag, insurance, language, marital_status, race, diagnoses_name, procedure_name, drug, is_elderly, long_stay_flag, death_within_24h_flag, has_diabetes, has_cardiac_issues, readmission_flag, icu_type_encoded, age_capped, age_group.

Target variables: Death_flag: Binary value (0/1) for classification purposes for an important on which tasks can lead to mortality

Log_los: Continuous variable that is used for regression on clinical procedures and how they affect LOS.

To assess model performance and generalizability, the dataset was randomly split into training, validation, and testing subsets using a 60/20/20 ratio, ensuring balanced representation across diverse patient types and outcomes.

Stratification based on the death_flag was applied to preserve class balance between survivors and non-survivors across all dataset splits.

The data split was not time-based, as the primary objective was to understand clinical procedures and patient outcomes rather than to forecast future trends.

By implementing a random yet stratified sampling approach, the model is robustly evaluated while minimizing the risk of data leakage.

The final dataset comprises 131 records with 30 features (8 numeric and 22 categorical) encompassing demographic, clinical, and derived variables. Length of stay (LOS) was partitioned using a 60/20/20 split, with no missing values present.

Section 2: Modeling

2.1 Modeling Strategy

The modeling approach incorporates both classification and regression techniques to address two key outcome variables. Classification is applied to predict patient mortality, using the death_flag variable to identify factors most strongly associated with death during an ICU stay. Regression analysis is used to model LOS, aiming to uncover variables that are highly correlated with extended hospitalization. Understanding these relationships enables the ICU department to target modifiable factors that may help reduce LOS. By integrating both classification and regression, the analysis captures variables that influence critical outcomes (mortality and LOS), allowing for a more comprehensive strategy to improve patient care and optimize resource utilization. Specifically, LOS modeling supports efforts to streamline hospital operations, while mortality prediction aids in identifying high-risk patients who may benefit from enhanced clinical attention.

The selected algorithms for modeling LOS include Ordinary Least Squares (OLS) Regression, Lasso, Ridge, and Random Forest Classifier. OLS was employed to identify linear relationships between LOS

Pipeline

Data cleaning

1. Import the raw dataset
2. View raw data. If the dataset is within the scope then the dataset is used. If it isn't then it is removed.
3. View kept datasets. Check each variables for missing and duplicates.
4. After view the missing variables and duplicates, drop any variables that will not be used.

Feature engineer

5. create new features to be used in analysis

Aggregation & transformation

6. Aggregate data into on dataset
7. Encode and transform data

Final Prep

8. Final split:Splitting the data for testing purposes.
9. Export data

and the input variables. Lasso introduced regularization and performed variable selection by penalizing less informative features, thereby reducing the risk of overfitting. Ridge regression addressed multicollinearity among predictors while also mitigating overfitting. Finally, the Random Forest Classifier was used to capture complex interactions and latent multicollinearity that may not have been detected by the linear models.

A 60/20/20 split for training, validation, and testing will be applied. The primary evaluation metric is the p-value threshold of 0.05, used to identify statistically significant predictors. While supplementary metrics such as R^2 and RMSE will be assessed to provide additional context on model performance, the core objective is to isolate variables that offer actionable insights for improving outcomes within the ICU department.

No computational issues are anticipated, as the dataset is clean and manageable, consisting of 131 records and 30 features. The data has been partitioned into training, validation, and testing subsets, further simplifying the analysis and ensuring efficient execution.

Model evaluation will prioritize statistical significance to guide stakeholders in making informed, targeted decisions. By identifying variables that are strongly correlated with key outcomes (LOS and mortality) we can recommend actionable clinical strategies. These insights support more effective ICU resource utilization, with the dual aim of reducing LOS and improving patient survival.

2.2 Model Development

The baseline models selected for this analysis were Ordinary Least Squares (OLS) regression for predicting length of stay (LOS) and logistic regression for mortality classification. These interpretable models served as benchmarks against more flexible algorithms such as Ridge, Lasso, and Random Forest.

For LOS prediction, Lasso and Ridge regression were chosen as the primary models due to their ability to regularize coefficients and manage multicollinearity. Lasso additionally performs variable selection by penalizing less informative features, while Ridge stabilizes estimates in the presence of correlated predictors.

Mortality classification was primarily handled using the Random Forest Classifier, which excels at capturing nonlinear relationships and identifying the most influential variables associated with patient outcomes.

Prior to modeling, all categorical variables were carefully reviewed. String-based categories, such as those in the age_group,features (e.g., young adult, adult, elderly), were label encoded to ensure compatibility with the modeling algorithms and prevent runtime errors.

Baseline models were run without hyperparameter tuning. However, for the primary models, Lasso and Ridge regression, GridSearchCV was employed to optimize hyperparameters systematically. This approach reduced manual trial-and-error, enhanced reproducibility, and minimized model bias by leveraging cross-validation to evaluate each parameter configuration.

All models executed successfully in under 30 seconds on a standard laptop, demonstrating computational efficiency.

In the OLS regression analysis for LOS, the only variable that met the statistical significance threshold (p-value < 0.05 at a 95% confidence interval) was the number of drugs administered. This suggests that increased medication usage is associated with longer ICU stays.

For mortality classification, two variables met the significance criteria: number of drugs administered (p-value = 0.035) and age_capped (p-value = 0.026). These findings indicate that older patients and those receiving a higher number of medications are more likely to experience mortality during their ICU stay. These insights can inform clinical decision-making and resource allocation strategies aimed at improving patient outcomes.

2.3 Model Evaluation and Comparison

Among the four regression models evaluated for predicting LOS, Ordinary Least Squares (OLS), Lasso, Ridge, and Random Forest, the Random Forest model demonstrated the strongest performance. It achieved the lowest root mean square error (RMSE = 0.54) and the highest coefficient of determination ($R^2 = 0.43$), indicating better fit and more accurate predictions. In comparison:

- OLS: RMSE = 0.65, $R^2 = 0.14$
- Lasso: RMSE = 0.64, $R^2 = 0.18$
- Ridge: RMSE = 0.61, $R^2 = 0.25$

These results suggest that Random Forest Regression is the most effective model for capturing the underlying patterns in LOS data.

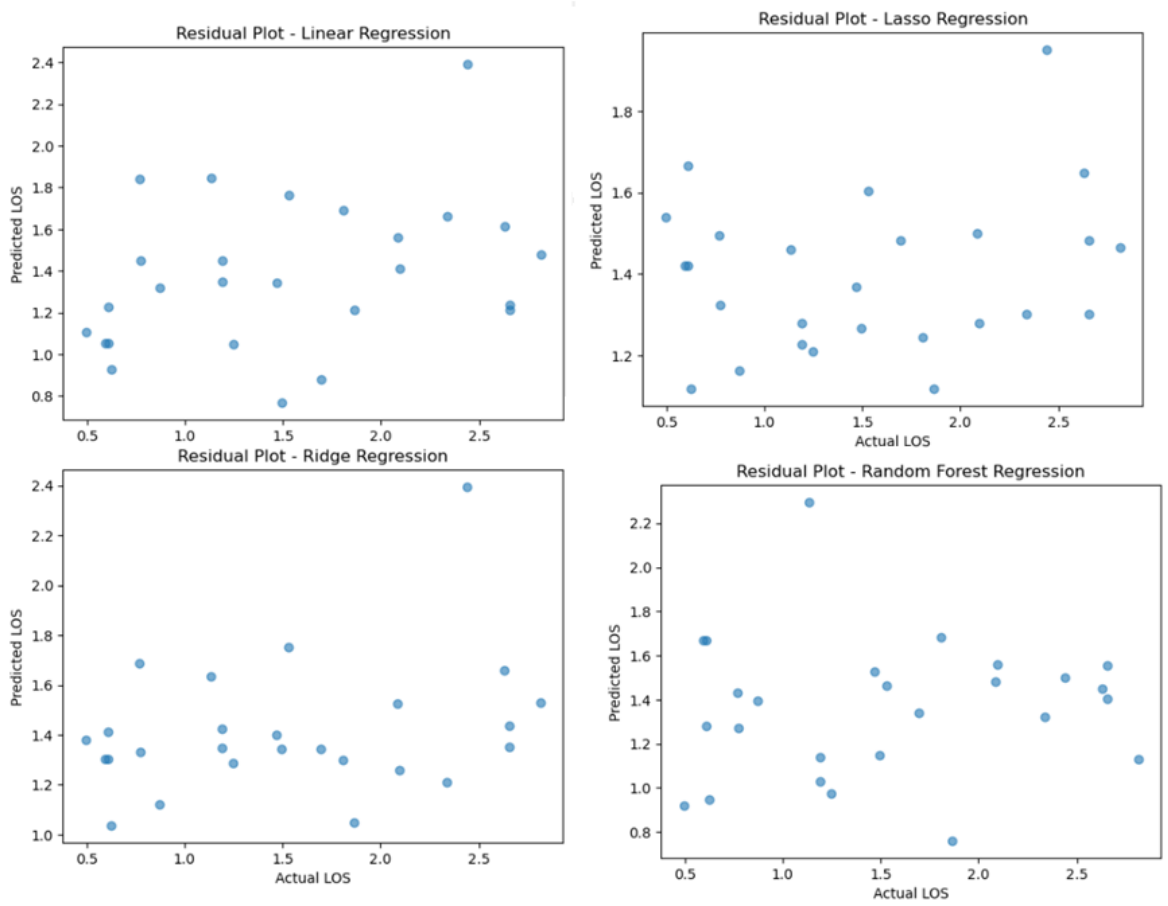
For mortality prediction, two models were assessed: logistic regression and Random Forest Classifier. The evaluation was based on the Receiver Operating Characteristic Area Under the Curve (ROC AUC)

score, which measures classification performance. Logistic regression yielded a ROC AUC of 0.60, only slightly better than random chance, while the Random Forest Classifier achieved a ROC AUC of 0.81, indicating strong predictive capability. Based on this metric, Random Forest Classifier is the preferred model for mortality classification.

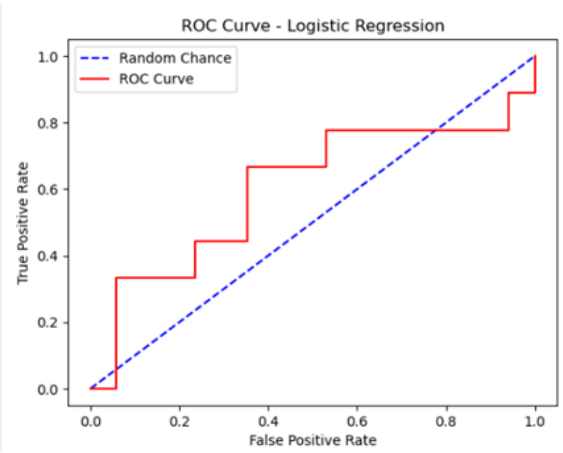
Despite Random Forest's superior performance, all regression models struggled to accurately predict higher LOS values. A consistent trend was observed: as actual LOS increased, model accuracy declined. Notably, Random Forest produced one significant outlier, predicting a LOS of 2.3 when the actual value was closer to 1. While OLS, Lasso, and Ridge performed better on extreme values, Random Forest was more effective at capturing average LOS patterns.

Below is a list of the Models, their characteristics and trade-offs:

- **OLS Regression**: Highly interpretable and straightforward, but limited by its inability to handle multicollinearity and nonlinear relationships, often leading to underfitting.
- **Lasso Regression**: Offers regularization and feature selection, reducing overfitting and multicollinearity. However, it requires careful tuning of the alpha hyperparameter, best handled via GridSearchCV to ensure consistency and efficiency.
- **Ridge Regression**: Manages multicollinearity effectively and retains all features, but lacks feature selection, as all coefficients remain non-zero.
- **Random Forest Regression**: Captures nonlinear relationships and variable interactions automatically, delivering improved predictive performance. However, it is computationally intensive and does not provide easily interpretable coefficients.
- **Logistic Regression**: Easy to interpret with clear odds ratios, but limited in predictive power, especially for complex patterns.
- **Random Forest Classifier**: Offers superior predictive accuracy and a higher ROC AUC score, though it requires more computational resources and lacks interpretability compared to logistic regression.



The mortality analysis was relatively straightforward. Logistic regression struggled to accurately predict outcomes, as reflected in its ROC curve performance. In contrast, the Random Forest model demonstrated stronger predictive capability, with its confusion matrix showing significantly better identification of true positives and true negatives.



Robustness was evaluated by applying multiple modeling techniques and comparing key statistical metrics, including RMSE, R^2, and ROC AUC scores. This multi-model approach ensured that the insights derived were data-driven rather than model-dependent. Consistency across models further reinforced the reliability of the findings. For instance, regression models for length of stay (LOS) consistently highlighted the number of drugs administered as a significant predictor, while classification models for mortality identified both drug count and patient age as influential variables. The recurrence of these predictors across different modeling strategies suggests that their impact is not an artifact of model bias, but rather a genuine association with patient outcomes.

By adhering to a 95% confidence interval and a p-value threshold of < 0.05, the analysis focused on variables with statistically significant effects on LOS and mortality. The number of drugs administered emerged as a key factor influencing both outcomes. Increased medication usage can prolong LOS due to the time required for administration (e.g., IV drips) and metabolic processing, and may also elevate the risk of adverse drug interactions, contributing to higher mortality. Age was another critical variable, with older patients showing increased susceptibility to mortality during ICU stays. Recognizing these associations enables clinicians to tailor care strategies—such as closer monitoring and more cautious prescribing—for high-risk patients, ultimately improving outcomes and optimizing ICU resource utilization.

2.4 Model Selection and Recommendations

Based on performance across both classification and regression tasks, the Random Forest model is the recommended approach moving forward. It consistently outperformed other models in predictive

accuracy, making it the most reliable choice for capturing patterns in the data. Its ability to closely approximate actual outcomes both for mortality and length of stay (LOS) was a key factor in its selection.

The primary criterion emphasized during model evaluation was predictive fidelity: how well the model could replicate real-world outcomes. In clinical contexts, this is especially critical. For example, a high rate of false positives (predicting death for patients who survive) could lead to unnecessary interventions and resource misallocation. While false negatives also carry risk, the priority was to minimize incorrect predictions that could compromise care quality. Random Forest's superior performance in this regard reinforced its selection.

Despite being the top-performing model, Random Forest Regression still exhibited limitations. Its RMSE of 0.54 and R^2 of 0.42 indicate moderate predictive power, but not ideal accuracy. A lower RMSE and higher R^2 would have increased confidence in its reliability. Additionally, the 60/20/20 train/validation/test split may have constrained model performance; a larger training set such as an 80/20 split could potentially yield better results.

Another concern lies in the interpretability and actionability of key predictors, particularly the number of drugs administered. While this variable showed strong associations with both LOS and mortality, it may be confounded by underlying factors such as the number of diagnoses or severity of illness. Patients with more complex conditions naturally receive more medications, which could skew the variable's perceived impact. This raises questions about whether drug count is a truly independent, actionable feature or a proxy for broader clinical complexity.

Conclusion:

After thorough data cleaning and evaluation across multiple modeling approaches, the Random Forest model emerged as the most effective for both regression and classification tasks. It consistently outperformed other models based on key performance metrics (RMSE, R^2, and ROC AUC) demonstrating strong predictive capability.

Across both analyses, the number of drugs administered surfaced as a significant predictor, influencing both length of stay (LOS) and mortality outcomes. Additionally, patient age was identified as a key factor in mortality prediction, though it did not show a strong association with LOS.

Moving forward, efforts should focus on reducing unnecessary drug administration to help minimize LOS and improve survival rates. Simultaneously, targeted strategies to enhance care for older patients in the ICU could further reduce mortality risk and optimize resource utilization. These insights provide actionable direction for improving clinical outcomes and operational efficiency within the ICU setting.