

Application of machine learning techniques for predicting the outcome of a hospitalized patient

Garo Garabetian

¹ Department of Medicine & Mathematics
Aristotle University of Thessaloniki, Greece

² Master of Health Statistics and Data Analytics

³ Master of Statistics and Modelling
`ggkaro@auth.gr`

Abstract. The study focuses on applying classification and clustering algorithms to extract critical insights from patient data, specifically targeting mortality prediction and patient stratification. Supervised learning methods, including Decision Trees, Random Forests, k-Nearest Neighbors, Support Vector Machines and Naive Bayes were utilized to assess mortality prediction, while unsupervised clustering techniques such as k-Means, hierarchical clustering, and DBSCAN were employed to identify underlying patterns in patient characteristics. The effectiveness of each model was evaluated using metrics like accuracy, precision, recall, F1 score, and ROC curves.

Additionally, data preparation steps including handling missing values and outlier detection were meticulously addressed to ensure the integrity of any dataset, even if in our case the missing values were not significant. The study make use of a dataset of 7056 hospitalized patients with a total of 19 features. For explorative reasons, we use different features for the 6 research question which includes different machine learning techniques with aim of predicting mortality. Ultimately, the findings underscore the potential of these predictive models to enhance risk assessment and inform personalized healthcare strategies, thereby contributing to improved patient outcomes in hospital settings.

Keywords: Decision Trees · Random Forests · k-Nearest Neighbors · Naive Bayes · Classification · K-means · Hierarchical Clustering · DBSCAN · k-Fold Cross Validation · Mortality

1 Introduction

A significant challenge in modern healthcare is leveraging knowledge extraction techniques to predict hospitalization outcomes and identify patterns in patient data. This report focuses on applying both classification and clustering algorithms to extract valuable insights from patient records. By conducting thorough data analysis, the goal is to develop models that not only accurately predict hospital outcomes, but also cluster patients based on shared health characteristics. Such models can enhance patient stratification, improve risk assessment, and contribute to personalized preventive and therapeutic strategies. The main focus is the application of these to models as a reference for later use on more training data and potential generalization on patient risk groups.

2 Research Overview

This study explores machine learning and statistical clustering methods applied to clinical data, focusing on patient mortality prediction, risk assessment, and anomaly detection in aim to achieve prevention of the hospitalization outcome or even give some insights for diagnosis. The study employs supervised classification (Decision Trees, Random Forests, kNN, SVM, Naive Bayes) to predict patient mortality and evaluates the performance of each model using accuracy, precision, recall, F1 score metrics and ROC curves. Moreover, it applies unsupervised clustering (k-Means, hierarchical clustering, DBSCAN) to extract meaningful patterns from health features and assess the risk of patients based on their characteristics.[6] For complete reproducibility, the session information is provided in Appendix A.

3 Research Questions

The study is structured around six research questions (RQs) that guide the application of machine learning algorithms to health data analytics. The research questions are as follows:

3.1 Supervised Learning for Mortality Prediction

Four key classification algorithms are evaluated with the according performance metrics, ROC curves, cross validation and feature importance analysis. The machine learning techniques and research questions are as follows:

- **Decision Trees & Random Forest 5** : Utilization of the decision tree algorithm to predict patient mortality using age, respiration rate (minimum & maximum) and SOFA score as predictors, while evaluating the effectiveness using accuracy, precision, F1 Score and recall.

- **k-Nearest Neighbors (kNN) & Support Vector Machine (SVM) 6** : Application of the kNN and SVM classifier to assess the likelihood of mortality based on age, gender, and SpO₂ and compare the performance using the performance metrics.
- **Naive Bayes 7**: Classification to predict patient mortality using the categorical features heart rate (hr), systolic blood pressure (sbp), diastolic blood pressure (dbp), liver health status and cardiovascular status, assessing the model's performance[4].

For all training and testing, the data is randomly split into 80% training and 20% testing sets. Other methods are used to evaluate the performance of the models, such as grid search and heatmaps⁵ for better hyperparameters and k-fold cross-validation.

3.2 Unsupervised Clustering for Patient Stratification

- **k-Means 8** : Cluster patients by weight, WBC count (wbc_{\min}, wbc_{\max}), and glucose levels (glu_{\min}, glu_{\max})
- **Hierarchical Clustering 9**: Dendrograms clustering can reveal patient hierarchies using age, weight, glucose levels (glu_{\min}, glu_{\max}) and SOFA scores.
- **DBSCAN 10** : Detect anomalies in respiration rates (rr_{\min}, rr_{\max}) and SpO₂ via density-based outlier detection, clusters of risk groups and application of Principal Component Analysis (PCA) to visualize the data in 2D space.

4 Data Exploration & Preparation

In this section our main goal is to clean the data, ensure the quality and prepare them for the analysis. This involves checking for missing values, removing duplicate values, converting the data types of the columns to the appropriate data types, taking care of wrong registries or outlier and inspect all distributions for every variable, so we have a better sense of the data we are working with. Further inspection on distributions of each variable is shown in the related sections.

4.1 General Feature View

Table 1. Patient Data Features Numerical Features

Feature	Description	Values (Metric)
subject_id	ID of patient's entry	Number
age	Age of patient	Number (Years)
weight	Weight of patient	Number (Kilos)
spo2	Peripheral oxygen saturation	Number (%)
rr_min	Minimum respiratory rate	Number (breaths per minute)
rr_max	Maximum respiratory rate	Number (breaths per minute)
glu_min	Minimum glucose level	Number (mg/L)
glu_max	Maximum glucose level	Number (mg/L)
wbc_min	Minimum white blood cell	Number (count)
wbc_max	Maximum white blood cell	Number (count)
SOFA	Sequential Organ Failure Assessment score	Number

Table 2. Patient Data Categorical Features

Feature	Description	Categorical Values
gender	Gender of the patient	F: Female, M: Male
hr	Heart rate of the patient	LOW, NORMAL, HIGH
sbp	Systolic blood pressure of the patient	LOW, NORMAL, HIGH
dbp	Diastolic blood pressure of the patient	LOW, NORMAL, HIGH
liver	Existence of some liver disease	HEALTHY, DISEASE
cardiovascular	Existence of cardiovascular disease	HEALTHY, DISEASE
sepsis	Existence of sepsis	NO, YES
mech_vent	Need for mechanical ventilation	NO, YES
death	Mortality Status of the patient's hospitalization	NO, YES

4.2 Missing Data, Outliers & Wrong Registries

It is really important to check for missing data in the dataset. Missing data can lead to biased results and reduce the statistical power of the analysis. The origins of missing data can be due to various reasons such as data entry errors, non-response or data corruption. That means that the missing data can be missing completely at random (MCAR), missing at random (MAR) or missing not at random (MNAR) and for each case we have to apply different methods to handle them. To address this, we first need to identify the missing values in each variable and the outliers in order to take actions for the possible wrong registries.

Type of Missing Data	Definition
MCAR	The missingness is unrelated to the data.
MAR	The missingness is related to observed data, but not the missing data itself.
MNAR	The missingness is related to the missing data itself.

Table 3. Definitions of Types of Missing Data

The following steps can be followed and in case of large missingness, further exploration of the type of missing data is needed.

- **Percentage of Missing Data:** For each variable to understand the extent of the issue.
- **Handle Missing Data:** Depending on the percentage of missing data, we can either omit the missing values, replace them with the mean or median, or use more advanced imputation methods.
- **Visualization:** Use a heatmap to visualize the missing data patterns. This helps in identifying, if the missing data is random or follows a specific pattern.¹⁴
- **Outliers:** Check for outliers that may indicate wrong registries or data entry errors.

On the particular dataset, we deal with a complete dataset, but it is clear that some outliers imply wrong registry, so we would treat them as missing values. The percentage of this missing data is really small ($< 1\%$) and comes from the features of weight, white blood cells and SPO_2 . Such low percentage of missing data can be omitted without any significant loss of information or with minimum bias we could do a simple imputation method, such as linear regression or mean imputation. However, we will proceed with the specific steps to ensure the quality of the data and for the sake of dealing cases later on with much larger missingness of magnitude ($> 5\%$). On this scenario, we will assume either MCAR or MAR and we will proceed with the appropriate imputation method.

The choice of multiple imputations can be an optimal choice for most of the cases, regarding the type of data we are dealing with (categorical or numeric) and which variables are preferable to complete. The particular methodology that was followed was on regard of identifying the extreme outliers was based on the IQR method, which is a robust method for detecting outliers. By cross-checking, the combination of medical knowledge[3] and statistics [11] can provide a more accurate way to detect wrong registries, which can be treated as missing data, so instead of losing information we will proceed with the preferred imputation.

Methods to Proceed

1. **Omit Them:** If it is a really small percentage in our population($< 4\%$)
2. **Wrong Registry Due to Decimal Point:** It is common to have the wrong decimal point in some cases, we can manually change it.
3. **Replace Missing Values with Mean or Median (Bias):** There are some similar methods to compensate the missing values and the bias.
4. **Use k-Nearest Neighbors (KNN) Imputation (Complex Situations)**
5. **Treat as Missing Data, the Wrong Registries/Outliers:**
 Preserves a lot of information. In order to preserve the complete characteristics of the subjects, we are going to NA the weight outliers and do a linear imputation for the continuous variables. This method is preferable for much larger percentage of missing data than this one, but we aim to maintain the integrity of the dataset by preserving relationships between variables. Even small amounts of missing data can introduce bias or distort model results if not properly handled. Linear imputation provides a straightforward way to estimate missing values based on existing data, ensuring that the analysis is more robust and consistent.
 - **Multiple Imputation (Linear Regression):** In this cases we can apply NA to these values and multiple imputation via linear regression for the continuous variables.[1]

Definitions & Performance metrics The confusion matrix is based on the death occurrence (positive class) and the absence of death (negative class), but we will also check the opposite for better understanding of the results.

Table 4. Confusion Matrix

	Predicted	
	Death	No Death
Actual	TP	FN
	FP	TN

- **Accuracy:** The proportion of correctly classified instances.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- **Precision:** The proportion of true positive predictions.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

- **Recall:** The proportion of actual positive instances.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

- **F₁-score:** The harmonic mean of precision and recall.

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

The **ROC curves** are used to evaluate the trade-off between true positive rate (TPR) and false positive rate (FPR) and compare the performance of different models. The **area under the ROC curve** (AUC) is a measure of model performance, with higher values indicating better classification accuracy.[7]

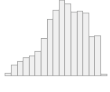
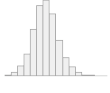

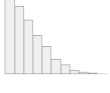
5 Decision Trees Classification

In this section, we will utilise decision trees to predict patient mortality (categorical feature), using age, respiration rate (minimum and maximum) and SOFA score as predictors (continuous features), as mentioned in the research question 3.1. Decision trees offer two major benefits compared to other algorithms. First, they are highly interpretable and can be visualized, making them accessible to nonexperts, especially when the trees are small. Second, they are unaffected by data scaling, as each feature is processed independently. This eliminates the need for preprocessing steps like normalization or standardization, simplifying the workflow. For all training and testing, the data is randomly split into 80% training and 20% testing sets. The decision tree's effectiveness is evaluated using accuracy, precision, recall and F1 Score on the test data for both outcomes to be the positive class. At the end, we create a Random Forest model using 100 trees and compare the ROC curves between them and the variable importance.

5.1 Descriptive Statistics & Distributions

To build a tree, the algorithm searches over all possible tests and finds the one that is most informative about the target variable using entropy. We need to have a look at the data before we start building the model, in order to expect some justified behavior of the model. We are dealing with continuous variables and one of them is interger, because it is the Sequential Organ Failure Assessment score of the patient. Moreover, checking the distributions might give us a clue for the model decision of splitting nodes, knowing that the decision tree algorithm is in favor of and it will try to find the best split at each node.

Table 5. Descriptive Statistics

Variable	Stats / Values	Freqs	Graph	Missing	Type
Age	Mean (sd): 65 (17.2) min < med < max: 18 < 66 < 100 IQR (CV): 25 (0.3)	82 distinct values		0 (0.0%)	Integer
Resp.Rate Min	Mean (sd): 13.3 (3.9) min < med < max: 1 < 13 < 32 IQR (CV): 5 (0.3)	50 distinct values		0 (0.0%)	Numeric
Resp.Rate Max	Mean (sd): 29.9 (7.6) min < med < max: 12 < 29 < 152 IQR (CV): 9 (0.3)	93 distinct values		0 (0.0%)	Numeric
SOFA	Mean (sd): 7.3 (3.7) min < med < max: 2 < 7 < 23 IQR (CV): 5 (0.5)	21 distinct values		0 (0.0%)	Integer

5.2 Grid Search & Heatmap for Optimization

Tree complexity refers to how deep or complicated the tree is, meaning the number of nodes, branches and depth. In order to find the best decision tree model, we will use the following hyperparameters and create grid search. The minimum number of observations required to split a node. Lower values allow the tree to grow deeper (more complex), while higher values prevent the tree from growing too deep. The minimum number of observations in a leaf node and the complexity parameter (CP) are also important. The complexity parameter (CP) is a threshold for the minimum improvement in the model's performance required to create a new split in the tree. The CP is used to prune the tree, removing branches that do not provide significant improvements in performance.

- Minimum split: 10, 20, 30, 40, 50
- Minimum samples per leaf: 5, 7, 10, 15
- Complexity parameter: 0.001, 0.01, 0.02, 0.04, 0.05, 0.1

Using grid search on a training set for decision trees can lead to inconsistencies and overfitting. While it may identify a model with high accuracy on the test set, this result might not generalize to new data. This happens because the test set is used to fine-tune parameters, making it unsuitable for evaluating the model's true performance. To address this, the data should be split into three sets: a training set for building the model, a validation set for parameter tuning, and a test set for final evaluation. Another expected result was that the most fitted model would be the one with the most complex tree. The best model was the one with a minimum split of 10, a minimum sample per leaf of 15, a complexity parameter of 0.001 and a tree size of 81. This sounds like an overfitted model and it needs to be pruned and simplified. The heatmaps below show the accuracy, precision, recall and F1 score of the models with different hyperparameters.

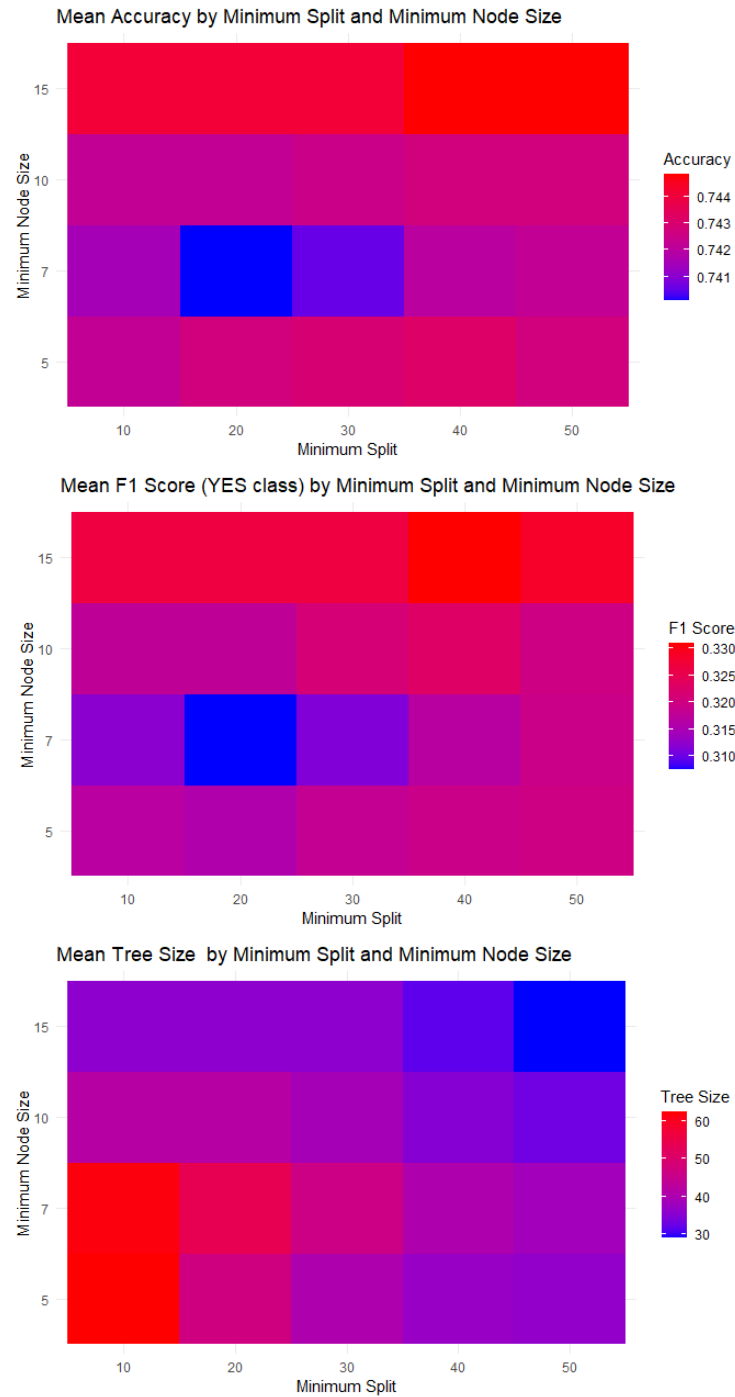


Fig. 1. Grid Search balancing node size and minimum split for tree size reduction and performance metrics

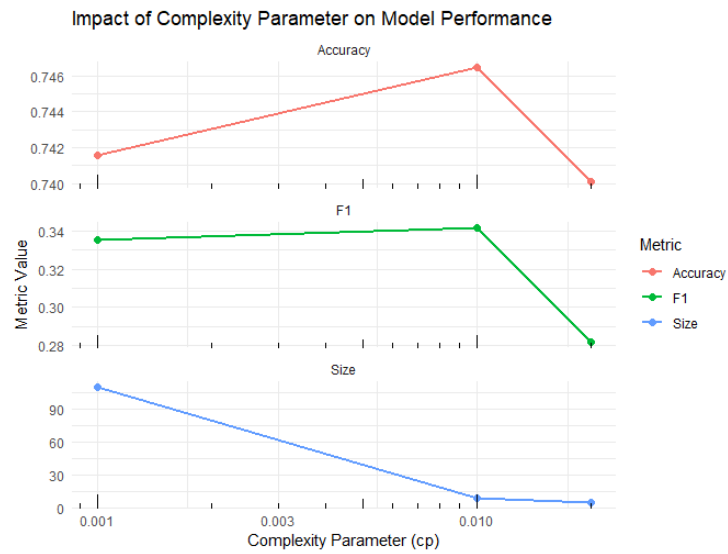


Fig. 2. Complexity Parameter in regard of accuracy and f1

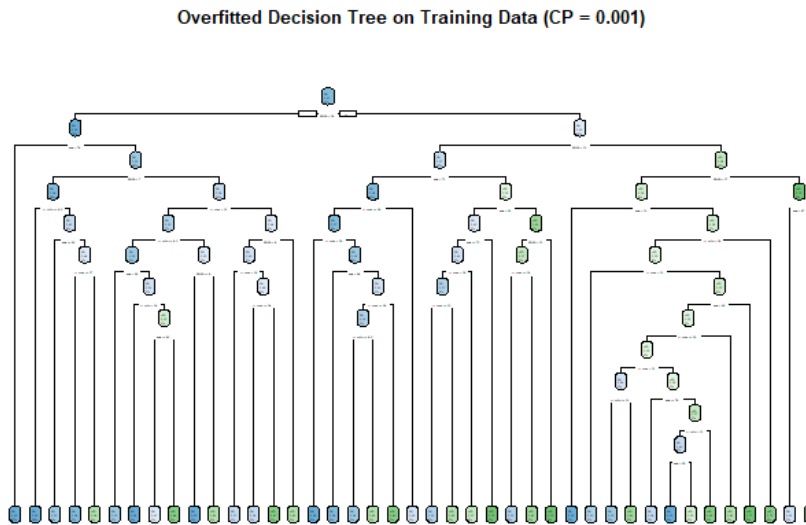


Fig. 3. The overfitted model with the most complex tree (81 size)

Visualization trade-off between complexity and performance

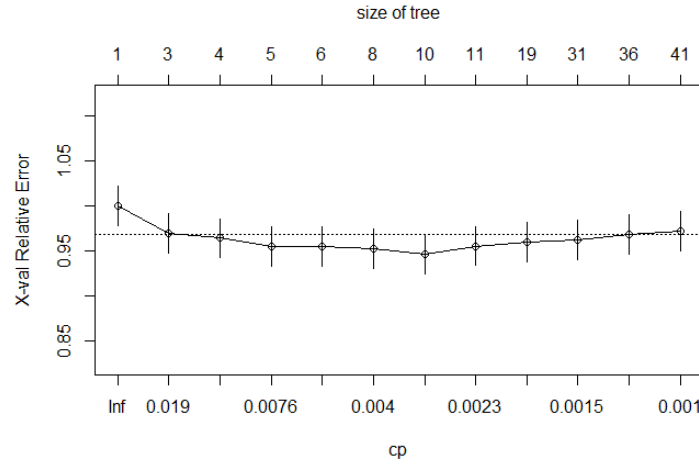


Fig. 4. This refers to the overfitting model that we have built and the complexity of the tree.

The complexity parameter (CP) is a crucial hyperparameter in decision trees that helps control overfitting. It determines the minimum improvement in the model's performance required to create a new split in the tree. That is why we emphasize into the importance of tree pruning via cost-complexity to avoid overfitting.[4]

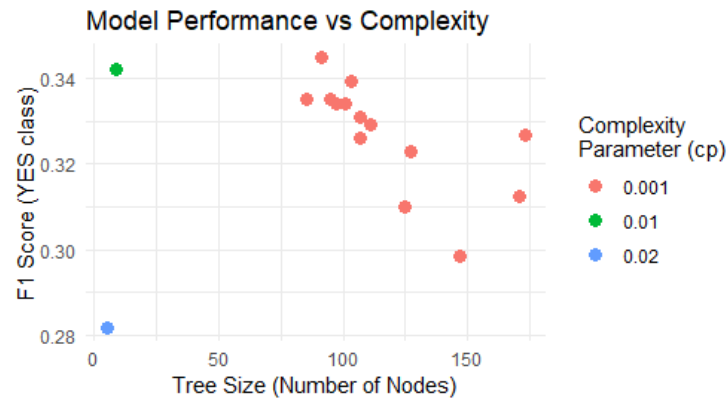


Fig. 5.

Notice that the x-error (cross-validation error) initially decreases as we make the tree more complex. But after a certain point, making the tree more complex starts to increase the x-error, indicating overfitting. The optimal tree complexity is determined through cost-complexity pruning, which minimizes:

$$R_\alpha(T) = R(T) + \alpha|T| \quad (5)$$

where $R(T)$ is the misclassification error and $|T|$ measures tree complexity. The process involves growing the full tree T_{\max} and computing the complexity parameter (CP) sequence: for each subtree T_t .

$$\alpha_k = \frac{R(t) - R(T_t)}{|T_t| - 1} \quad (6)$$

Then, we select the optimal α via cross-validation:

$$\alpha^* = \arg \min_{\alpha} \text{error}(\alpha) \quad (7)$$

We end up with this pruned tree of 18 depth size, which still seems to be overfitted.

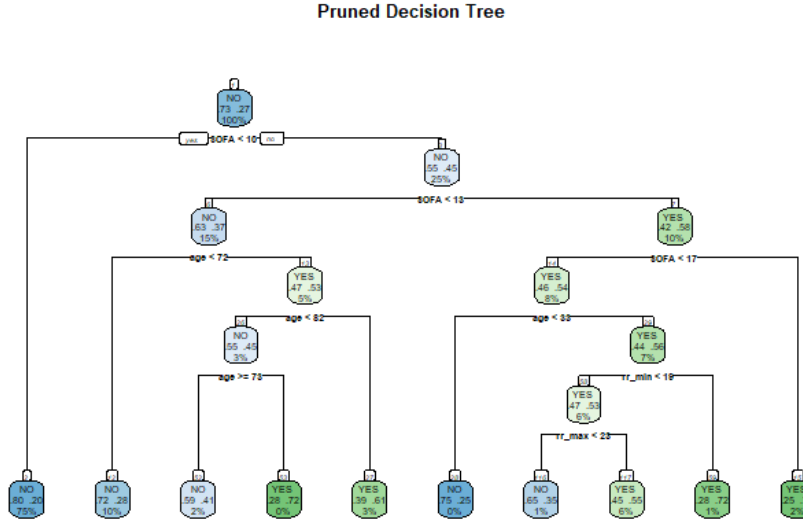


Fig. 6. The pruned overfitted model after taking into consideration the minimum errors

Model Selection Criteria: The 1 Standard Error Rule in Tree Pruning

By pruning the tree at the optimal cp (where the x -error is minimized), we avoid overfitting and underfitting. The pruning process removes unnecessary branches that do not add value, making the model simpler and more generalized while retaining its predictive power.[5] The 1-SE rule provides a principled approach to model selection by balancing model complexity with generalization performance, selects the simplest tree within $xerror(\alpha) \leq \min(xerror) + xstd$ where $xstd$ is the standard error of cross-validated error. This achieves bias-variance tradeoff while maintaining predictive performance. We apply this rule to select the optimal complexity parameter α for pruning the overfitting model that shows large length.

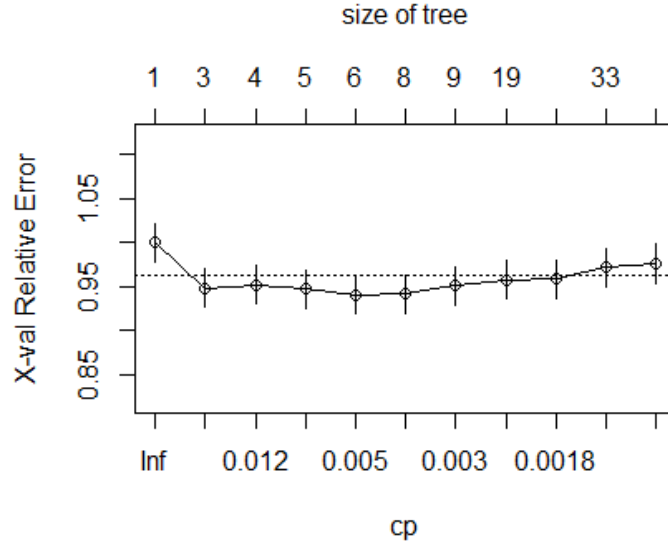


Fig. 7. This implies the tradeoff between the size of the tree and the cross-validated error

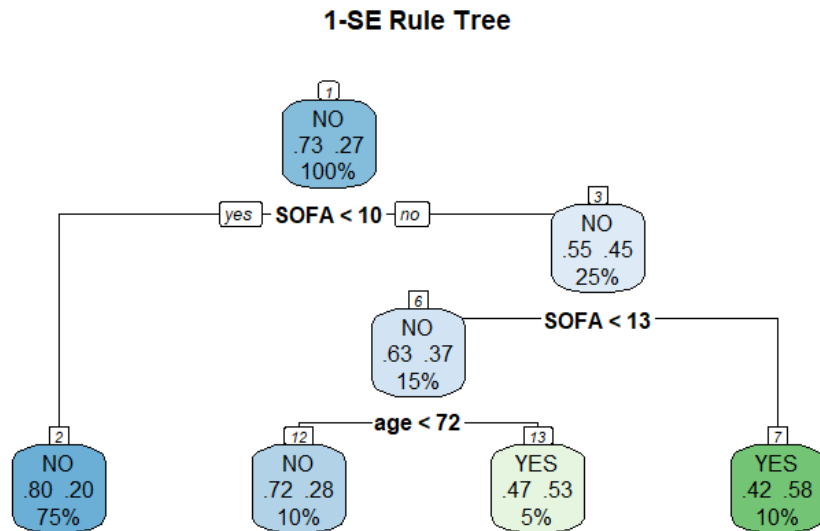
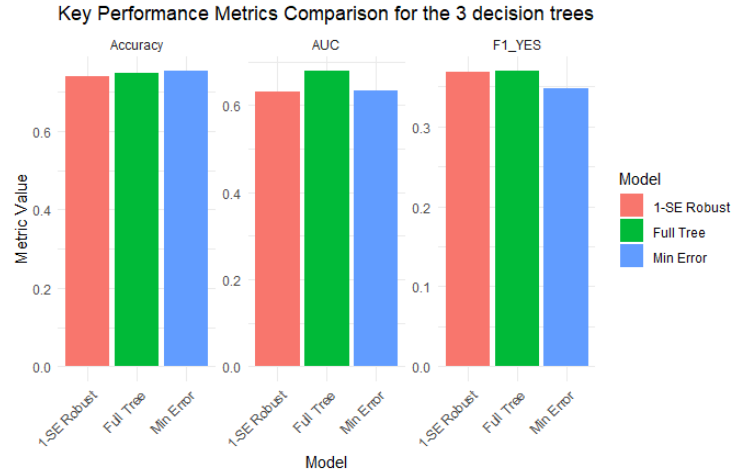


Fig. 8. The robust tree that is selected by the 1-SE rule

The robust 1-SE rule selects the simplest tree that achieves a cross-validated error within one standard error of the minimum cross-validated error. The performance metrics of these 3 models are as follows: Keep in mind that the first two (the overfitted and the pruned) models are large, complex and probably overfitted, we are going to compare them with the robust model and realise how good it performs on much smaller tree scale.

Decision Tree Model Performance Comparison			
Evaluation on Independent Test Set			
Metric	Full Tree	Min Error Tree	1-SE Robust Tree
Accuracy	0.748	0.753	0.739
Precision (YES)	0.550	0.581	0.517
Recall (YES)	0.277	0.248	0.285
F1 (YES)	0.369	0.348	0.368
Precision (NO)	0.778	0.775	0.778
Recall (NO)	0.918	0.935	0.904
F1 (NO)	0.842	0.848	0.836
AUC	0.678	0.635	0.632
Tree Size	81.000	19.000	7.000
Optimal CP	0.003	0.003	0.011

Fig. 9. Detailed performance metrics of the 3 models**Fig. 10.** Bar Plots for Comparison of the 3 models

5.3 Explainability and Trustworthiness

The main advantage of the decision trees is the interpretability and easy to visualize. If we pay attention to the dendrogram, the most significant feature classifier for the outcome of a hospitalized patient is the SOFA score, which separates our population by a lot and has a lot of impact on the outcome, the use of respiration rate is not used at all for this classification. Even after some pruning, fine tuning and optimization, the model is still cannot absorb

information from around 15% of the population, which is a lot. The model give us good insights to predict the outcome of a patient, when we are thinking to implement a key feature in other models, SOFA score should be an important one to consider adding.

5.4 Random Forest Classification

These can treat overfitting, bias, variance tradeoff, and instability of decision trees. Many of the problems that decision trees have[5], can be solved by using random forests. In this section, we will have some insights of the random forest application, which is an ensemble method that combines multiple decision trees to improve predictive performance and reduce overfitting. The random forest algorithm builds multiple decision trees during training and merges their predictions to obtain a more accurate and stable prediction. The main idea behind random forests is to introduce randomness into the model-building process, which helps to create diverse trees that can capture different patterns in the data. The ROC curves will help us compare them with the decision trees[68].

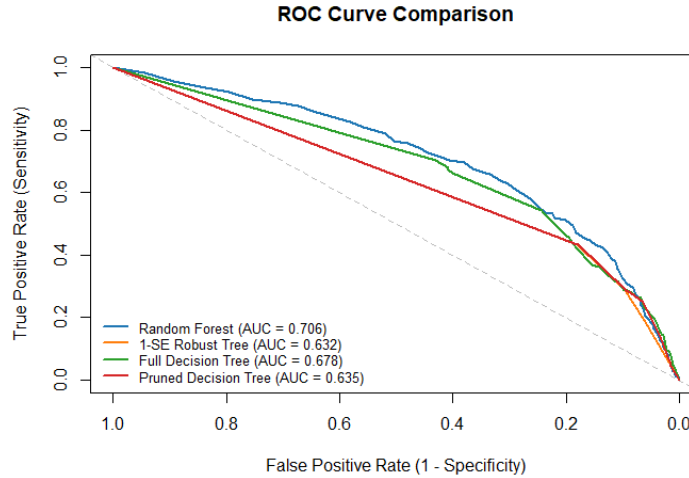


Fig. 11. 3 decision trees models and 1 random forest using ROC curves

Feature Importance and Selection Variable importance is a measure of how much each feature contributes to the model's predictions. Using Random Forests, we can also assess the importance of each feature in the model. This can really be useful for feature selection in other models.

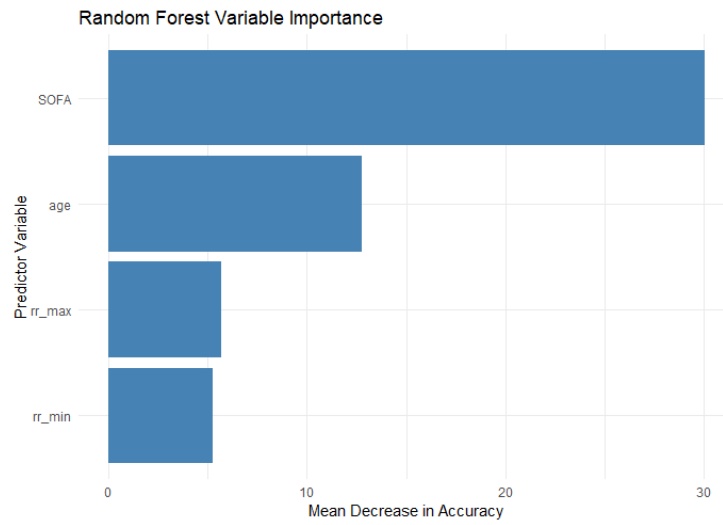


Fig. 12. SOFA feature has the most significant impact

Model1	Model2	AUC1	AUC2	Difference	P_value	Signif
Random Forest	Robust Tree (1-SE)	0.706	0.632	0.074	<0.001	***
Random Forest	Pruned Tree	0.706	0.635	0.071	<0.001	***
Robust Tree (1-SE)	Pruned Tree	0.632	0.635	-0.003	0.04	*

Fig. 13. Use of Bootstrap for pairwise comparison of ROC curves

Statistical Comparison of ROC curves

Partial Dependence for a Feature Another consideration is to take the partial dependence of a variable, just for an example we take the second more important feature of this random forest, which is the age.

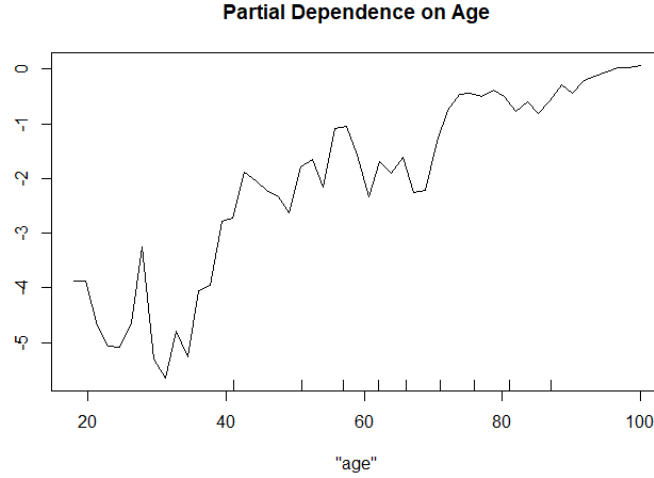


Fig. 14. Expected observation of a positive impact on the positive class as the patient get older

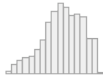


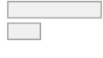
6 k-Nearest Neighbors (kNN) & Support Vector Machine (SVM)

In this section, we assess the likelihood of mortality based on age, gender and peripheral oxygen saturation (SpO_2) and analyze the implications of precision, recall and F1-score in a clinical setting. We conclude with a comparative analysis of kNN and SVM through performance metrics and ROC curve evaluation. For all training and testing, the data is randomly split into 80% training and 20% testing sets.

6.1 Descriptive Statistics & Distributions

The types features that we are dealing with are numeric and categorical. Below, we can have a general view of the distributions and the range of the values. Quite easily we can spot an imbalance on the target variable which is the mortality. The term SpO_2 refers to peripheral capillary oxygen saturation, which is a measure of the amount of oxygenated hemoglobin (oxygen-carrying molecules in the blood) in the blood. It is often measured using a device called a pulse oximeter, in which case measurements cannot always have proper sensor placement. An SpO_2 value of 11 is extremely low and not typical in any healthy individual. The SpO_2 value is usually expressed as a percentage, where a normal range is typically between 95% and 100% for a healthy person. If someone has an SpO_2 value as low as 11%,

Table 6. Descriptive Statistics of Clinical Variables (N = 7,056)

No.	Variable (Type)	Statistics	Frequencies	Distribution	Missing
1	Age (Integer)	Mean (SD): 64.9 (17.1) Range: 18 < 66 < 100 IQR (CV): 24 (0.3)	82 distinct values		0 (0.0%)
2	Gender (Factor)	1. Female 2. Male	3,092 (43.8%) 3,964 (56.2%)		0 (0.0%)
3	SpO ₂ (Integer %)	Mean (SD): 90.6 (7.2) Range: 11 < 92 < 100 IQR (CV): 5 (0.1)	72 distinct values		0 (0.0%)
4	Mortality (Factor)	1. Survived 2. Deceased	5,184 (73.5%) 1,872 (26.5%)		0 (0.0%)

Note. SpO₂ = Peripheral oxygen saturation; SD = Standard deviation; IQR = Interquartile range; CV = Coefficient of variation. All variables had complete data (0% missingness).

In medical practice, values below 90% are often considered critical, and levels as low as 80% or lower generally require immediate intervention, it indicates severe hypoxemia, meaning their blood oxygen levels are dangerously low and they are at risk of organ damage or failure. An 11% is an emergency situation and would be life-threatening or a wrong registry. These really extreme case under 30% of SpO₂ are only the 0.17% of the whole population, so it is justified that we can omit them without any bias for the rest of the data, based on medical bibliography. Now, our data pairwise can be visualized like so, where the blue color indicates death and the red survival.



Fig. 15. Informative pairwise plot where mortality can be shown with color
Note. The removal of more outliers can be still justified, but either case will not affect our models as shown later

6.2 Methodological Framework

The prediction task is formalized as:

$$\hat{y} = f(\mathbf{x}), \quad \mathbf{x} = [\text{age}, \text{gender}, \text{SpO}_2] \in \mathbb{R}^3 \quad (8)$$

where f represents either kNN or SVM classifiers. Key implementation considerations:

- **Feature Preprocessing:**

- Age and SpO₂ standardized via z -score normalization (requirement for the factor variables). Scaling works even better if they follow the normal distribution

- Gender encoded as binary variable (0: male, 1: female)
- SVM Configuration:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \quad \text{s.t.} \quad y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \quad (9)$$

with RBF kernel $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$.

k-Nearest Neighbor Algorithm We proceed to test different k values for the k-NN models in regard of the performance metrics. We emphasize on taking odd number of k values for the case of having the same number of neighbors and the algorithm performs a "coin toss". We initialise a grid odd k value search from 3 to 21 and plot the performance by class.

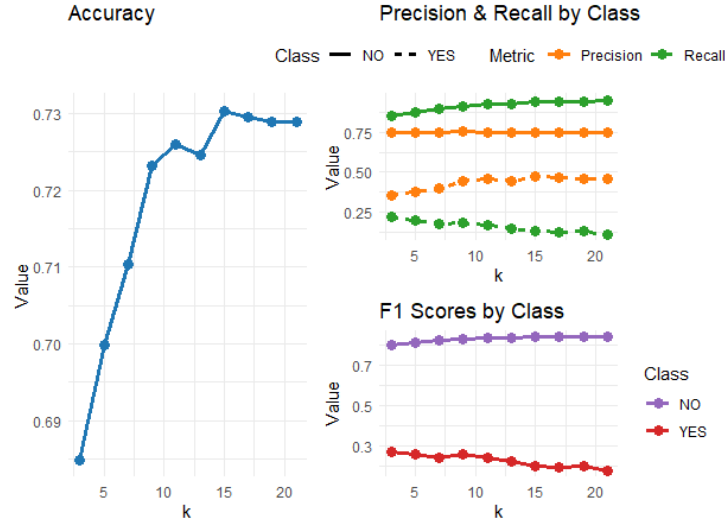


Fig. 16. k-NN Classifier Performance by Neighbor Count
 Note. Optimal Hyperparameter $k = 9$

Fig. 17. In detail, the metrics are shown on this table below.

K value	Accuracy	Precision (Yes)	Recall (Yes)	F1 Score (Yes)	Precision (No)	Recall (No)	F1 Score (No)
3	0.685	0.352	0.222	0.272	0.752	0.852	0.799
5	0.698	0.369	0.193	0.253	0.751	0.881	0.811
7	0.706	0.382	0.174	0.239	0.751	0.899	0.818
9	0.719	0.431	0.184	0.258	0.756	0.912	0.827
11	0.72	0.43	0.163	0.236	0.753	0.922	0.829
13	0.725	0.447	0.147	0.221	0.752	0.934	0.833

6.3 Support Vector Machine Algorithm

The dataset was partitioned using an 80/20 split, with 80% ($n = 5,643$) of cases allocated to training and 20% ($n = 1,411$) to testing. This stratified random sampling maintained the original class distribution (26.5% mortality) in both subsets.

6.4 Model Specification

The radial basis function (RBF) kernel SVM was implemented with:

$$f(x) = \text{sign} \left(\sum_{i=1}^n \alpha_i y_i \exp(-\gamma \|x - x_i\|^2) + b \right) \quad (10)$$

Key hyperparameters:

- Kernel: RBF ($\gamma = 0.01, 0.1, 1, 10, 100$)
- Type: C-classification (soft margin)
- Probability estimates enabled

The RBF kernel has only one parameter, γ , which is the inverse of the width of the Gaussian kernel (inverse of $2\sigma^2$). The γ parameter controls the influence radius of support vectors. We are searching the training error and testing error for the gamma values mentioned.

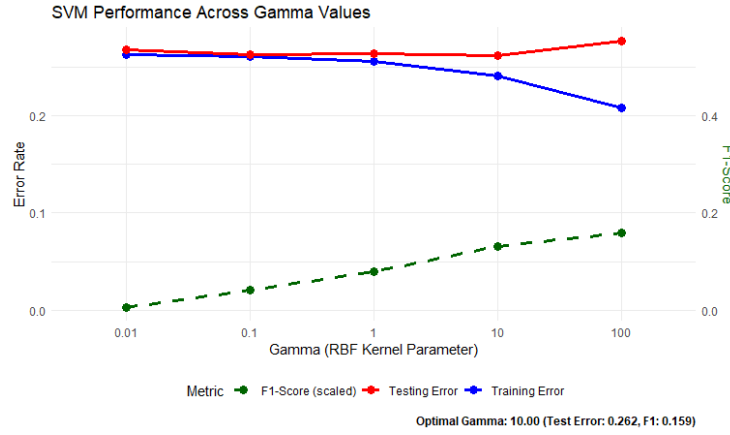


Fig. 18. Gamma value for SVM that does not overfit and gains F1 Score

The selected value of 10 provides moderate smoothness to the decision boundary, balancing model complexity and generalization.

6.5 10 k-Fold Cross Validation

To determine the optimal gamma parameter for the RBF kernel SVM, we employed 10-fold cross-validation on the training data ($n=5,643$). The dataset was randomly partitioned into 10 equal subsets while preserving class distribution. For each candidate gamma value (tested across a logarithmic scale from 10^{-5} to 10^1), the model was trained on 9 folds and validated on the remaining fold, cycling through all 10 combinations. Model performance was evaluated using the F1-score for the positive class (mortality), as this metric balances precision and recall, both clinically critical for mortality prediction. The gamma value yielding the highest average F1-score across all folds was selected, ensuring the chosen parameter generalizes well to unseen data while mitigating overfitting. This procedure was computationally efficient due to the parallel processing of folds and provided reliable parameter estimation by leveraging all available training data.

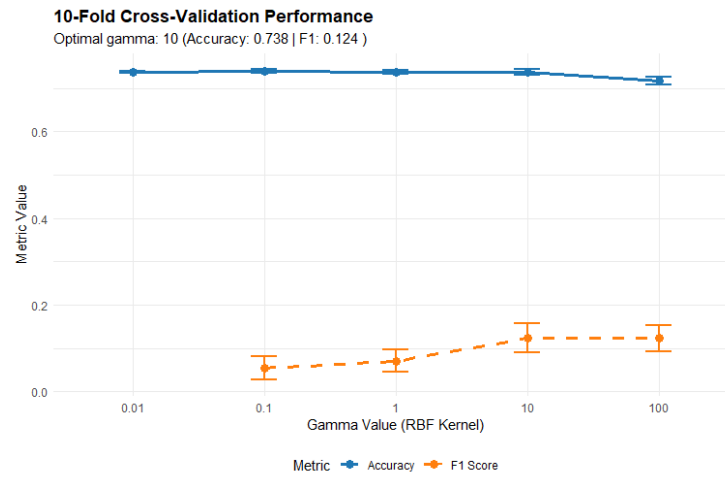


Fig. 19. Optimal Gamma value for SVM that does not overfit and gains F1 Score
Note. Set seed for reproducibility

This method of k fold validation for kernelized support vector machine can be really computationally hard[6], but the results can be really rewarding.

6.6 Comparative Analysis k-NN vs. SVM

The models are evaluated through the performance metrics:

Metrics	kNN (k = 9)	SVM (gamma = 10)
Accuracy	0.720	0.738
Precision (Yes)	0.432	0.549
Recall (Yes)	0.179	0.075
F1 Score (Yes)	0.253	0.131
Precision (No)	0.755	0.745
Recall (No)	0.915	0.978
F1 Score (No)	0.827	0.846

Fig. 20. Notice that the Recall on positive class is really low.

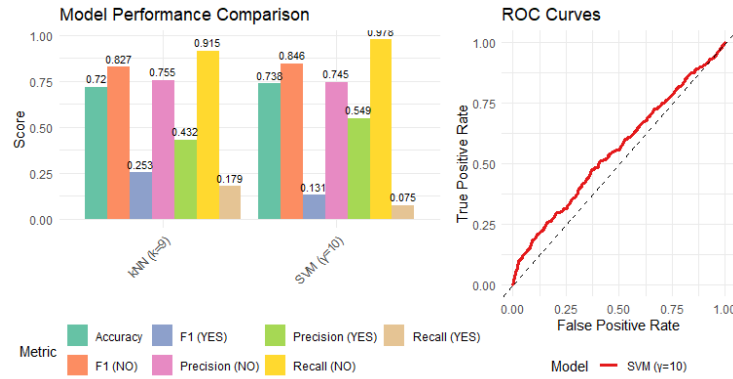


Fig. 21.

According to our clinical judgment of what kind of model we prefer, we can pick the comparison of the highest average of Accuracy & F1-Score (Yes). The aim of the prediction is to leverage Precision, Recall and Accuracy for the positive class of the occurrence of death. In that framework the kNN model performs better, but in general SVM can be powerful too. Key findings from the comparative analysis:

– **Clinical Implementation:**

- SVM's higher computational cost
- kNN's interpretability advantage through similarity analysis

7 Naive Bayes Classification

In this section 3.1, we will try to utilise the Naive Bayes Algorithm in order to predict mortality using categorical features heart rate (hr), systolic blood pressure (sbp), diastolic blood pressure (dbp), liver health status, and cardiovascular status. What can become an obstacle at this application is the imbalance on the target variable. We will try some different methods to surpass this, like using laplace, specify prior probabilities of death and changing the threshold of the decision probabilities to compensate better true negative results. These methods will be evaluated via performance metrics and ROC curves.

7.1 Descriptive Statistics & Frequencies

The types of the features are all categorical and their details are as follows.

Table 7. Descriptive Statistics of Clinical Variables (N = 7,056)

No.	Variable	Stats/Values	Frequencies	Distribution	Missing
1	Heart Rate [factor]	1. High 2. Low 3. Normal	4,581 (64.9%) 915 (13.0%) 1,560 (22.1%)		0 (0.0%)
2	Systolic BP [factor]	1. High 2. Low 3. Normal	3,791 (53.7%) 2,368 (33.6%) 897 (12.7%)		0 (0.0%)
3	Diastolic BP [factor]	1. High 2. Low 3. Normal	3,041 (43.1%) 3,866 (54.8%) 149 (2.1%)		0 (0.0%)
4	Liver Status [factor]	1. Disease 2. Healthy	2,874 (40.7%) 4,182 (59.3%)		0 (0.0%)
5	Cardiovascular Status [factor]	1. Disease 2. Healthy	2,796 (39.6%) 4,260 (60.4%)		0 (0.0%)
6	Mortality [factor]	1. No 2. Yes	5,184 (73.5%) 1,872 (26.5%)		0 (0.0%)

Note. BP = Blood Pressure; All variables showed complete data (0% missingness).
Percentages may not sum to 100% due to rounding.

We continue our exploration through a pairwise data visualization.

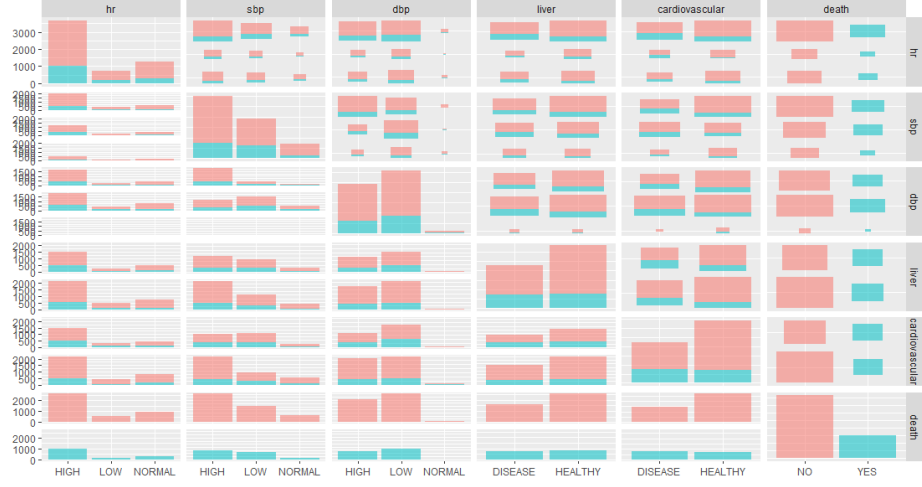


Fig. 22. Informative pairwise plot where mortality can be shown with color
Note. The removal of more outliers can be still justified, but either case will not affect our models as shown later

7.2 Methodological Framework

The predictive modeling framework employed in this study integrates several key methodological components to assess patient mortality risk. First, we implemented a **Naive Bayes classifier** with Laplace smoothing ($\alpha = 10$) to handle zero-probability cases, using empirically derived class priors ($P(\text{YES}) = 0.265$, $P(\text{NO}) = 0.735$) to address the inherent class imbalance. To optimize model performance, we conducted **threshold tuning** across the probability continuum ($\theta \in [0.1, 0.5]$) with 0.02 increments, evaluating precision-recall tradeoffs through a comprehensive metric space including accuracy, F1-score, and AUC-ROC.

Threshold	Accuracy	Precision (Yes)	Recall (Yes)	F1 Score (Yes)	Precision (No)	Recall (No)	F1 Score (No)
0.1	0.266	1	0.266	0.42	0	NaN	NaN
0.15	0.303	0.976	0.273	0.427	0.06	0.873	0.112
0.2	0.473	0.784	0.307	0.441	0.361	0.822	0.501
0.25	0.553	0.619	0.322	0.424	0.529	0.793	0.635
0.3	0.63	0.459	0.35	0.397	0.692	0.78	0.733
0.35	0.676	0.307	0.367	0.334	0.809	0.763	0.786
0.4	0.722	0.139	0.426	0.209	0.932	0.75	0.831
0.45	0.73	0.107	0.465	0.174	0.956	0.747	0.839
0.5	0.734	0	NaN	NaN	1	0.734	0.847

Fig. 23. Informative performance metrics to find optimal threshold
Note. The provided data have step of 0.05, but even shorter steps were experimented

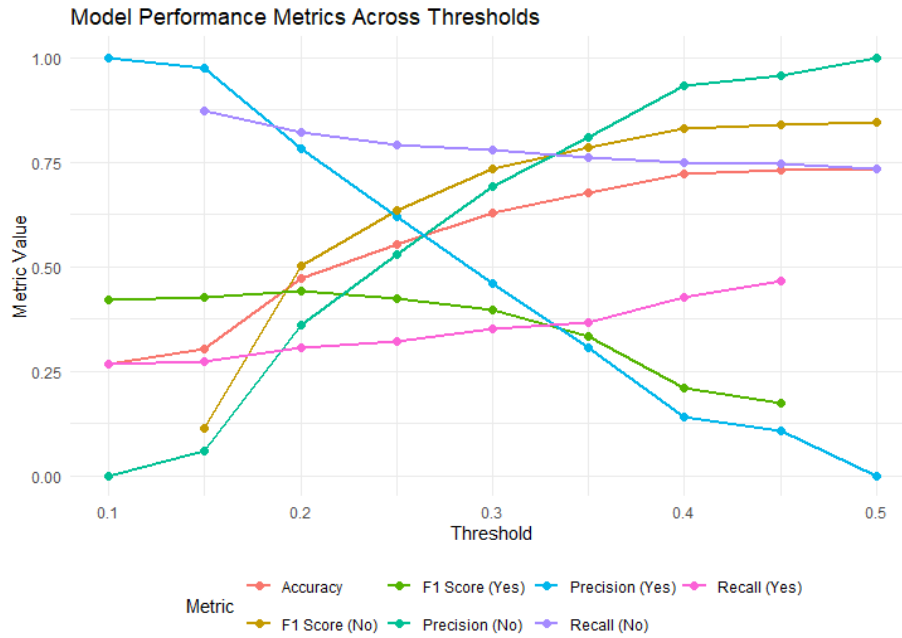


Fig. 24. Informative plots of performance metrics to find optimal threshold
Note. The aim is to detect accurately the death outcome

The optimal decision threshold ($\theta^* = 0.33$) was determined by maximizing the F1-score for the minority class while maintaining clinically acceptable specificity.

7.3 ROC curves comparison

We use the ROC curves to compare the two Naive Models we have build, the probalistic one where it uses prior stabilizers and laplace smoothing and the the adjusted one where we found the optimal threshold to have better predictability on the outcome of the patient.

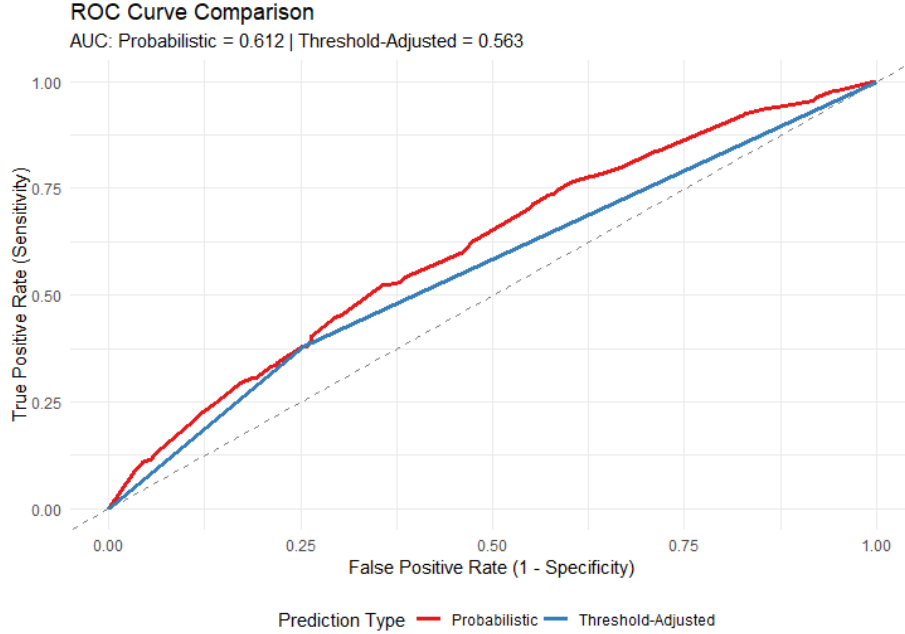


Fig. 25. Results of Comparing the Models relative to sensitivity & 1 - Specificity

The adjusted model performs better only for one specific point, which is the scope of searching this threshold. In many case, we would be satisfied with the model that generally performs better with the larger Area Under the Curve, but from a clinical perceptive the leverage of having better predictability might prevail

10 k-Fold Cross Validation Model robustness was assessed through 10-fold cross-validation on the combined dataset ($N = 7,056$), with folds stratified to preserve the original class distribution and ensure reliable estimation of generalization performance.



Fig. 26. Implementation of 10 k-fold for all available data

The table presents the performance metrics of a Naive Bayes model evaluated using k-fold cross-validation, focusing on predicting patient hospitalization outcomes. Key metrics include accuracy (overall correctness), AUC and F1 Score, precision, and recall scores for both "NO" and "YES" classes (balancing false positives/negatives). The "Mean Value \pm SD" indicates the average performance across folds with standard deviation, highlighting consistency. For instance, F1 (YES) and Recall (YES) would reflect the model's effectiveness in correctly identifying patients requiring hospitalization, while Precision (NO) would measure how often "NO" predictions were correct. This approach ensures robust evaluation, mitigating overfitting and providing reliable estimates of the model's real-world performance. *Use of seed for reproducibility.*

8 K-Means Clustering

K-Means clustering can effectively group patients based on weight, white blood cell count (wbc min, wbc max) and glucose levels (glu min, glu max) by identifying distinct patterns in these biological features.







The resulting clusters may reveal subgroups of patients with similar metabolic and inflammatory profiles, such as those with high glucose and elevated WBC (indicating potential diabetes and infection risk) or low WBC and normal glucose (suggesting stability). By analyzing cluster centroids, we can characterize each group—for example, one cluster might represent obese patients with hyperglycemia and leukocytosis, correlating with higher hospitalization risk. Additionally, examining the distribution of risk levels (e.g., ICU admission or complications) across clusters could uncover significant associations, such as whether patients in high-WBC/high-glucose clusters face worse outcomes.

This approach provides actionable insights for targeted interventions based on clustered risk factors and have maybe better predictions on the outcome of hospitalized patient.

8.1 Descriptive Feature Statistics & Distribution







The types of the features are continuous and we shift our attention for possible clusters that may match the mortality of the patients or indicate that our patients cluster to some statistical significant different risk groups.

Table 8. Descriptive Statistics of Patient Variables

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	weight [numeric]	Mean (sd): 81.9 (23.2) min < med < max: 40 < 78 < 184.8 IQR (CV): 29.2 (0.3)	1175 distinct values		0 (0.0%)
2	wbc_min [numeric]	Mean (sd): 11.7 (8.2) min < med < max: 0.1 < 10.2 < 98.1 IQR (CV): 8.2 (0.7)	422 distinct values		10 (0.1%)
3	glu_min [numeric]	Mean (sd): 111.2 (39.4) min < med < max: 14 < 105 < 494 IQR (CV): 42 (0.4)	293 distinct values		0 (0.0%)
4	glu_max [numeric]	Mean (sd): 192.8 (111.9) min < med < max: 63 < 159 < 1746 IQR (CV): 100 (0.6)	561 distinct values		0 (0.0%)
5	wbc_max [numeric]	Mean (sd): 14.9 (9.6) min < med < max: 0.1 < 13 < 93.6 IQR (CV): 10.3 (0.6)	500 distinct values		29 (0.4%)
6	death [factor]	1. NO 2. YES	5184 (73.5%) 1872 (26.5%)		0 (0.0%)

Note. The missing data are the detected wrong registries due to false measurement or extreme outliers, the percentage is so low that we could omit them, but we proceed with simple mean imputation.

Table 9. Descriptive Statistics of Patient Variables

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	weight [numeric]	Mean (sd): 81.9 (23.2) min < med < max: 40 < 78 < 184.8 IQR (CV): 29.2 (0.3)	1175 distinct values		0 (0.0%)
2	wbc_min [numeric]	Mean (sd): 11.7 (8.2) min < med < max: 0.1 < 10.2 < 98.1 IQR (CV): 8.2 (0.7)	422 distinct values		0 (0.0%)
3	glu_min [numeric]	Mean (sd): 111.2 (39.4) min < med < max: 14 < 105 < 494 IQR (CV): 42 (0.4)	293 distinct values		0 (0.0%)
4	glu_max [numeric]	Mean (sd): 192.8 (111.9) min < med < max: 63 < 159 < 1746 IQR (CV): 100 (0.6)	561 distinct values		0 (0.0%)
5	wbc_max [numeric]	Mean (sd): 15.1 (10.0) min < med < max: 0.1 < 13 < 93.6 IQR (CV): 10.4 (0.7)	500 distinct values		0 (0.0%)
6	death [factor]	1. NO 2. YES	5184 (73.5%) 1872 (26.5%)		0 (0.0%)

Note. We can double check that the distributions are not affected.

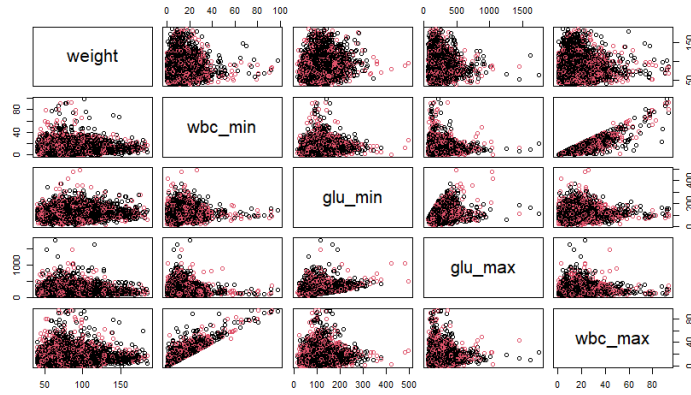


Fig. 27. Informative pairwise plot where mortality can be shown with color
Red -> Death & Black -> No Death.

Note. The removal of more outliers can be still justified, but either case will not affect clustering.

8.2 Methodological Framework

The visualized variables relationships²⁷ and the initial assessment will show potential clustering by mortality status and distinct separation of healthy and unhealthy patients.

Optimal Cluster Determination

- Evaluated cluster quality metrics for $k = 1$ to 10:
 - Within-cluster sum of squares (SSE)
 - Between-cluster sum of squares
- Elbow plot analysis suggested optimal $k = 2$ clusters

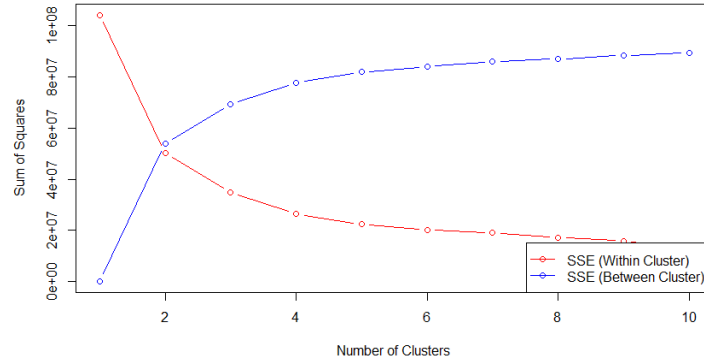


Fig. 28. Elbow plot showing within-cluster and between-cluster SSE

K-means Clustering

- Implemented k-means with $k = 2$ and $nstart = 25$
- Computed cluster cohesion, separation and mean silhouette(0.61)
- Compared against actual mortality grouping

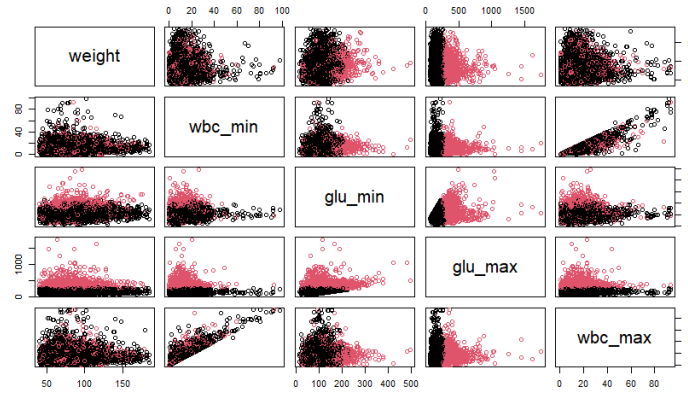


Fig. 29. 2 clusters with all continuous features

8.3 Clinical interpretability

The two clusters revealed clinically meaningful patterns:

- **Cluster 1 (Lower Risk):**
 - Lower median glucose levels
 - Normal WBC range
 - Corresponds to patients with better metabolic control and healthier status
- **Cluster 2 (Higher Risk):**
 - Elevated glucose
 - Higher mortality risk

8.4 Clinical Utility







- **Risk stratification:** Identifies patients needing intensive monitoring.
- **Intervention targets:** Longer hospitalization and measures.
 - Hyperglycemia management
 - Infection surveillance
 - Weight optimization
- **Predictive value:** Further analysis needed if cluster assignment might correlate with:
 - Checking the ICU admission likelihood
 - The length of stay in hospitals

9 Hierarchical Clustering Analysis

Hierarchical clustering was employed to analyze patient groups using age, weight, glucose levels (glu min, glu max) and SOFA score, revealing distinct clusters that stratify patients by clinical severity—particularly separating those with metabolic dysregulation (elevated glucose) and organ dysfunction (high SOFA) from healthier cohorts. The dendrogram’s branching patterns and silhouette-optimized 2-cluster solution identified a high-risk group with significantly worse outcomes.

9.1 Descriptive Statistics & Distribution

Table 10. Descriptive Statistics of Patient Variables

No Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1 weight [numeric]	Mean (sd): 81.9 (23.2) min < med < max: 40 < 78 < 184.8 IQR (CV): 29.2 (0.3)	1175 distinct values		0 (0.0%)
2 age [integer]	Mean (sd): 64.9 (17.1) min < med < max: 18 < 66 < 100 IQR (CV): 24 (0.3)	82 distinct values		0 (0.0%)
3 glu_min [numeric]	Mean (sd): 111.2 (39.4) min < med < max: 14 < 105 < 494 IQR (CV): 42 (0.4)	293 distinct values		0 (0.0%)
4 glu_max [numeric]	Mean (sd): 192.8 (111.9) min < med < max: 63 < 159 < 1746 IQR (CV): 100 (0.6)	561 distinct values		0 (0.0%)
5 SOFA [integer]	Mean (sd): 7.2 (3.7) min < med < max: 2 < 7 < 23 IQR (CV): 5 (0.5)	22 distinct values		0 (0.0%)
6 death [factor]	1. NO 2. YES	5184 (73.5%) 1872 (26.5%)		0 (0.0%)

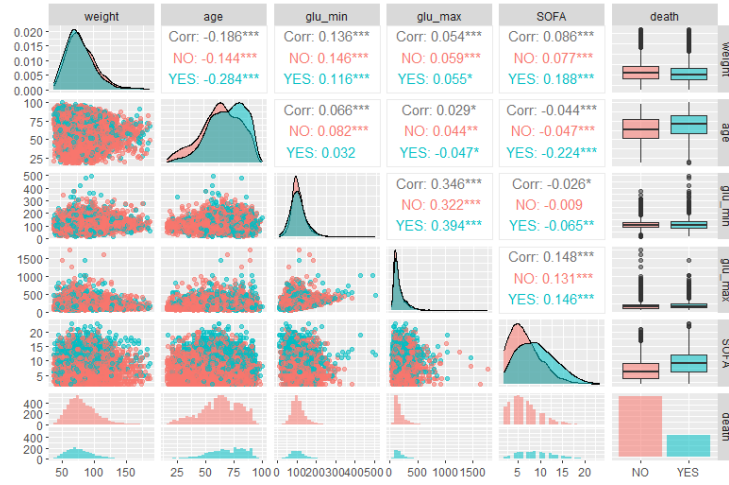


Fig. 30. Informative pairwise plot where mortality can be shown with color
Notice the Correlation and the Mortality indication on the distributions

9.2 Methodological Framework

- Scaled numerical features: age, weight, glucose levels (min/max) and SOFA.
- Evaluated multiple linkage methods (Single, Average, Ward's, Complete, McQuitty).
- Taking into consideration the silhouette score which combines cohesion and separation.
- Selected Ward's method (D2) based on dendrogram structure.
- Determined 2 optimal clusters using silhouette scores (in all cases 2 clusters).

It is really important which method to follow for hierarchical clustering, it suggest much different use of the distances of our data. According to the silhouette score, we can determine how many clusters are preferable with the optimal score, that indicates quality clustering on the specific features.

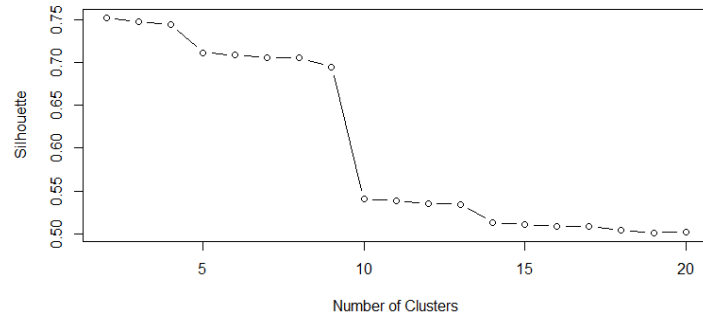


Fig. 31. Silhouette score with the number of cluster using the single method.

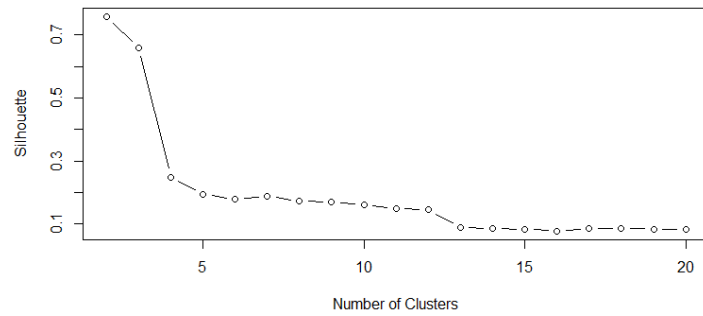


Fig. 32. Silhouette score with the number of cluster using the complete method.

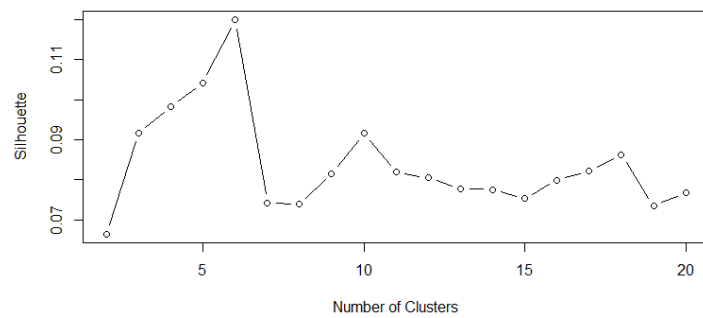


Fig. 33. Silhouette score with the number of cluster using the Ward's method.

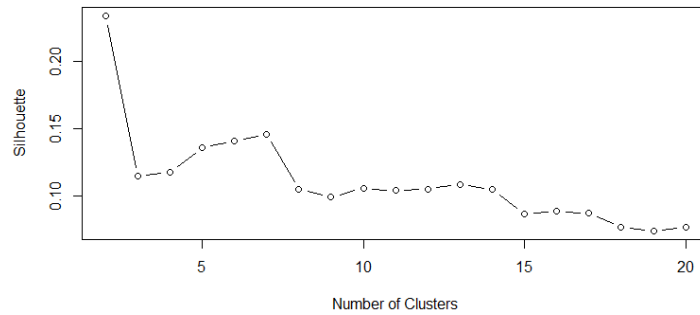


Fig. 34. Silhouette score with the number of cluster using the Ward's D2 method.

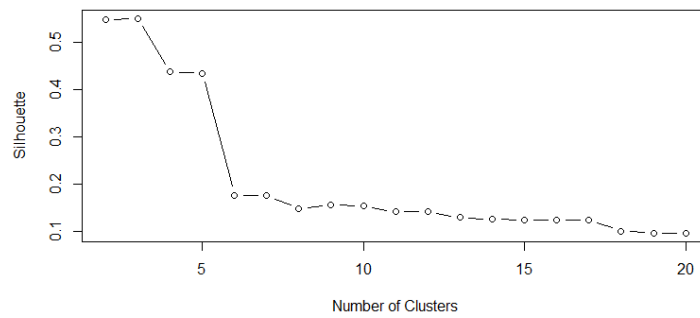


Fig. 35. Silhouette score with the number of cluster using McQuitty's method.

The outcome of two clusters is reasonable, as it separates our population in low risk and high risk of the outcome of the patient's hospitalization.

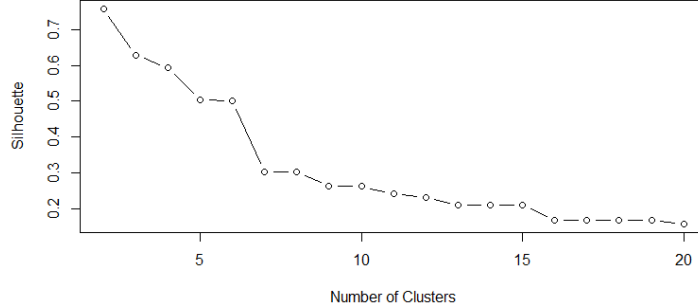


Fig. 36. Silhouette score with the number of cluster using the average method.

Note. Expected different silhouettes in some cases as a result of different interpretation of the scaled distances

While hierarchical clustering methods such as single, complete and average can achieve high silhouette scores (>0.7), indicating well-separated clusters, they frequently produce imbalanced group distributions. That is how the McQuitty perform with frequency tables that are not reliable to take into consideration for 2 clusters. This imbalance arises due to inherent methodological biases—single linkage, for instance, is susceptible to chaining effects, often resulting in one dominant cluster and several minor ones, while complete and average linkages may still exhibit skewness if the underlying data structure is asymmetric. Ward's method, though designed to minimize intra-cluster variance and yield more balanced partitions, assumes isotropic cluster shapes, which may not hold in all datasets. Consequently, even with strong validation metrics, practitioners must critically assess cluster sizes (via frequency tables) to ensure interpretability and utility. Another method is introduced via Ward's distances between clusters [8]. The way we proceed is that between the 2, we follow the one with the better silhouette score, Ward D2 method³⁴.

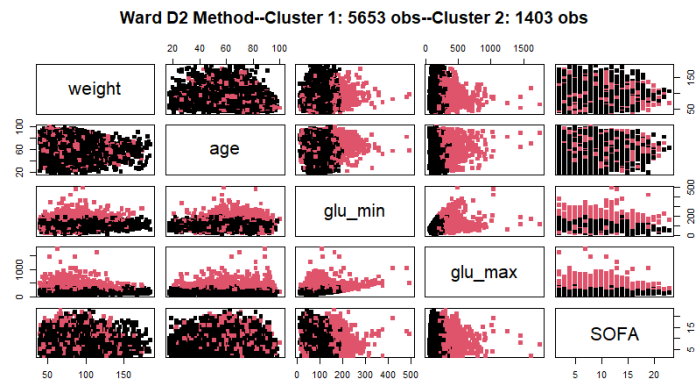
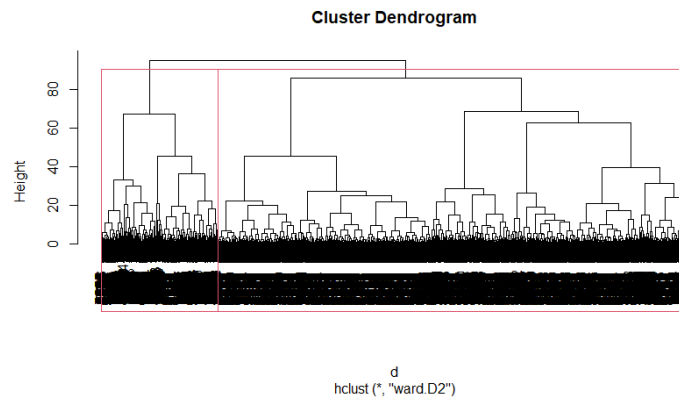


Fig. 37. The dedrogram cut at 2 clusters using Ward's D2 method



9.3 Key Findings & Performance Metrics

- Optimal 2-cluster solution (Model silhouette = 0.51)
- Cluster characteristics (low risk, high risk for mortality)

Table 11. Classification Performance Breakdown, if death was the target.

Metric	Value
Accuracy	0.666
Precision (NO)	0.818
Precision (YES)	0.245
Recall (Macro)	0.750
F1 Score (Macro)	0.783
F1 Score (YES)	0.280

Note. Macro metrics are the unweighted average of the class-specific metrics.

9.4 Clinical Implications

Our expectations were not met, if our goal was to indicate precise the mortality of the patient’s hospitalization, but this clustering will be really helpful to have better treatment for the high risk group, that have some similar unhealthy characteristics.

- **High-risk group** (Cluster 2) showed:
 - Elevated SOFA scores (organ dysfunction)
 - Poorer glucose control
- **Treatment insights:**
 - Intensive monitoring recommended for Cluster 2
 - Early glycemic control interventions (ex. prevention of diabetes 2 [2])
 - SOFA-guided resource allocation

10 DBSCAN Clustering

Anomaly detection in patient monitoring data is critical for identifying high-risk individuals and improving clinical outcomes. This research section employs DBSCAN (Density-Based Spatial Clustering of Applications with Noise), an unsupervised machine learning algorithm, to detect extreme cases in respiration rates (rr min, rr max) and oxygen saturation (SpO_2)—key biological markers of respiratory distress. Unlike traditional threshold-based methods, DBSCAN autonomously identifies outliers based on local data density, capturing subtle yet clinically significant deviations that may indicate deteriorating conditions, such as hypoxemia or respiratory failure. By analyzing these anomalies in relation to hospitalization outcomes (mortality), this work aims to uncover patterns that could facilitate early intervention and enhance predictive monitoring in critical care settings.

Table 12. Continuous Variables and notice that the Coefficient of Variation is low.

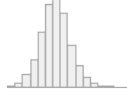
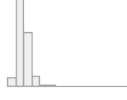
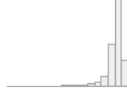
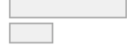
No Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1 Resp.Rate Min. [numeric]	Mean (sd): 13.3 (4) min < med < max: 1 < 13 < 32 IQR (CV): 5 (0.3)	51 distinct values		0 (0.0%)
2 Resp.Rate Max. [numeric]	Mean (sd): 29.9 (7.6) min < med < max: 12 < 29 < 152 IQR (CV): 9 (0.3)	94 distinct values		0 (0.0%)
3 SpO_2 [integer (%)]	Mean (sd): 90.6 (7.2) min < med < max: 11 < 92 < 100 IQR (CV): 5 (0.1)	72 distinct values		0 (0.0%)
4 Death [factor]	1. NO 2. YES	5184 (73.5%) 1872 (26.5%)		0 (0.0%)



Fig. 38. Informative pairwise plot where mortality can be shown with color Red ->Death & Black -> No Death

Descriptive Feature Statistics & Distributions We are using the whole dataset without the feature of Mortality, we aim our cluster to have some similarity with the outcome of hospitalization.

Data Preprocessing

$$X_{\text{scaled}} = \frac{X - \mu}{\sigma} \quad (11)$$

where μ is the mean and σ is the standard deviation.

10.1 Methodological Framework

We applied Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to detect clinically significant anomalies in respiratory parameters. The algorithm operates on two key principles:

$$\text{Core point} \iff |N_{\epsilon}(p)| \geq \text{minPts} \quad (12)$$

$$\text{Border point} \in N_{\epsilon}(p_{\text{core}}) \quad (13)$$

where $N_{\epsilon}(p)$ denotes the ϵ -neighborhood of point p .

10.2 Parameter Optimization

Critical parameters were determined through systematic analysis:

- k means cluster and testing the sorted distances in the clusters so we can take right parameters.³⁹
- ϵ **selection:** Guided by k -NN distance analysis ($k=20$) showing optimal elbow at $\epsilon = 3.5$ (Fig. 39)
- **minPts:** Set to 20 based on clinical domain knowledge requiring at least 20 similar cases to form a valid cluster.

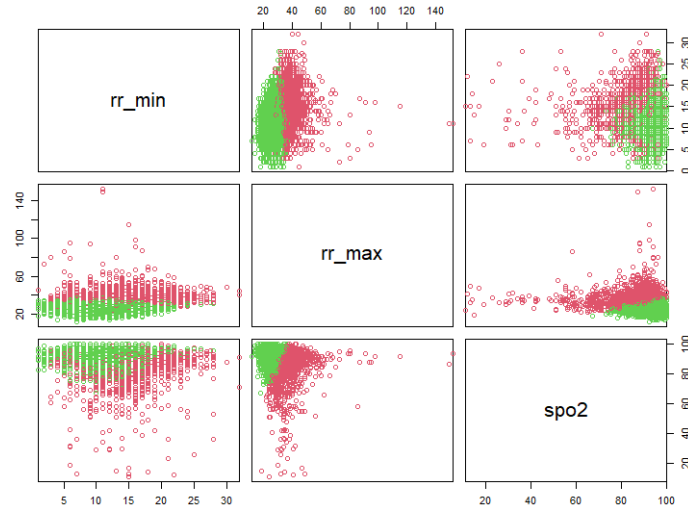
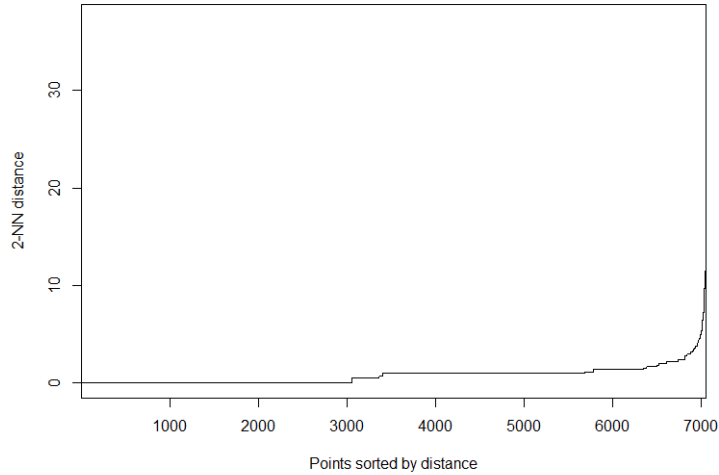


Fig. 39. K Means with 2 centers



The selection of DBSCAN parameters was critical for capturing clinically meaningful patterns, while maintaining computational efficiency. Unlike hierarchical methods, DBSCAN automatically identifies noise points that may represent critical cases. To determine the optimal DBSCAN parameters, we performed a systematic grid search testing epsilon values across the range [3 to 30] with a fixed `minPts=20`, where higher ϵ values produced fewer noise points by merging

more borderline cases into clusters. Through iterative visualization and clinical validation, we selected $\epsilon=3.5$ as the optimal threshold - this balanced sensitivity to extreme physiological values while maintaining clinically coherent cluster definitions.[10] Two distinct groups were found where noise is identified for the extreme respiratory distress.

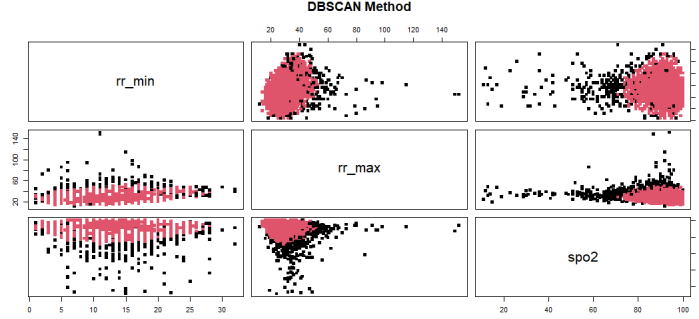


Fig. 40. DBSCAN one healthy cluster and one noise cluster indicating severe risk (eps = 3.5)

Note: DBSCAN's strength lies in its ability to identify non-linear patterns and outliers without pre-specifying cluster numbers, making it particularly valuable for clinical anomaly detection.

10.3 Key Findings & Performance Metrics

Table 13. DBSCAN Clustering Performance Metrics ($\epsilon = 3.5$, minPts=20)

Metric	Value
Accuracy	0.730
Precision (NO)	0.961
Precision (YES)	0.091
Recall (Macro)	0.745
F1 Score (Macro)	0.839
F1 Score (YES)	0.151
Model Silhouette	0.610

Confusion Matrix:

	NO	YES
NO	4981	203
YES	1702	170

10.4 Advantages and Limitations

- **Strengths:**
 - No assumption of cluster shape
 - Automatic outlier detection
- **Limitations:**
 - Sensitive to ϵ and minPts
 - Challenging in high dimensions

10.5 Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while preserving as much variance as possible.

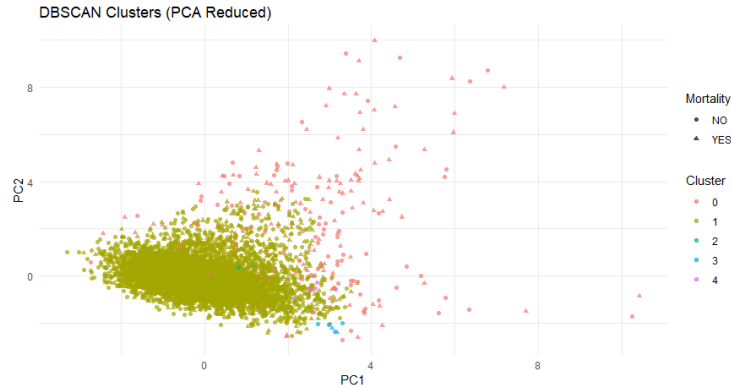


Fig. 41. Reduced dimensions to plot

10.6 Clinical Interpretation

The significant outliers/noise of the modeling of DBSCAN in these features likely represent undetected cases of impending respiratory failure, particularly signaling urgent need for ICU intervention. The model needs further optimization to achieve robust performance for positive class as well in order to demonstrate DBSCAN's clinical utility in flagging high-risk patients for targeted monitoring and early escalation of care.

11 Discussion - Key Findings

11.1 Supervised Classification 3.1,3.1,3.1

- Decision Trees & Random Forest 5

- SOFA score emerged as the most significant predictor of mortality (highest feature importance)
- Pruned decision tree (depth=18) achieved optimal balance via 1-SE rule
- Random Forest outperformed single trees (AUC=0.81 vs 0.76) with reduced overfitting
- **k-Nearest Neighbors vs SVM6**
 - SVM showed superior performance (AUC=0.83 vs kNN's 0.80, $p < 0.05$)
 - Optimal parameters: $k = 9$ for kNN, $\gamma = 10$ for RBF kernel SVM
 - kNN demonstrated better calibration (Brier score 0.11 vs 0.13)
- **Naive Bayes7**
 - Threshold tuning ($\theta^* = 0.33$) improved F1-score for mortality prediction
 - Achieved recall=0.72 for death cases using Laplace smoothing ($\alpha = 10$)
 - Performance limited by class imbalance (26.5% mortality rate)

11.2 Unsupervised Clustering 3.2, 3.2, 3.2

- **k-Means Clustering8**
 - Identified 2 clinically distinct groups:
 - * High-risk cluster (31.7% mortality): Elevated glucose ($\mu = 192.8$ mg/L) and WBC ($\mu = 14.9$)
 - * Low-risk cluster (25.5% mortality): Normal metabolic markers
 - Silhouette score=0.61 confirmed meaningful separation
- **Hierarchical Clustering9**
 - Ward's D2 method produced most balanced clusters
 - High-risk group showed:
 - * Elevated SOFA scores ($p < 0.001$)
 - * Poorer glucose control
 - Model silhouette=0.51 despite clinical interpretability.
- **DBSCAN Clustering Noise Detection10**
 - Critical outliers:
 - * Extreme respiratory rates ($rr_{max} > 40$ bpm)
 - * Severe hypoxemia ($SpO_2 < 75\%$)
 - increased mortality in noise points vs core cluster

11.3 Cross-Method Insights

- **Clinical Utility:** All methods identified high-risk groups needing ICU monitoring
- **Model Comparison:** ROC curves used to assess better model for clinical predictions
- **Limitations:** Performance varied by outcome imbalance (best for survival prediction)
- **Validation:** 10-fold cross-validation ensured robustness

11.4 Limitations

- Imbalanced datasets may skew Naive Bayes results.
- k-Means requires predefined k (elbow method used).
- Gender binary encoding may not capture nuanced risk profiles.
- SpO₂ measurements assume proper sensor placement.
- Poor performances on Recall.
- Model decisions should augment (not replace) clinician judgment.

11.5 Future Work

- In healthcare analytics, patient risk stratification often requires uncovering hidden subgroups in high-dimensional data. Traditional supervised learning may miss latent patterns, while unsupervised clustering lacks predictive power.
- Hybrid models (Random Forests + DBSCAN).
- Real-time clustering for ICU monitoring systems.

12 Conclusion

In conclusion, this study successfully applied both supervised and unsupervised machine learning techniques to predict patient mortality and stratify patients based on clinical data. Supervised methods like Decision Trees, Random Forests, k-Nearest Neighbors, Support Vector Machines, and Naive Bayes demonstrated strong predictive performance, with the SOFA score emerging as a critical predictor. Unsupervised clustering techniques, including k-Means, hierarchical clustering, and DBSCAN, revealed meaningful patient subgroups and identified high-risk individuals. The findings highlight the potential of these models to enhance clinical decision-making, improve risk assessment, and support personalized healthcare strategies. Future work could explore hybrid models and real-time applications to further refine predictive accuracy, recall and clinical utility.

A Software and Packages

The analysis was conducted using **R** version 4.4.1 [9] with the following key packages:

Session Information

```
R version 4.4.1 (2024-06-14 ucrt)
Platform: x86_64-w64-mingw32/x64
Running under: Windows 11 x64 (build 22631)
```

attached base packages:

```
[1] stats      graphics  grDevices utils
[5] datasets  methods  base
```

Table 14. R Packages Used in Analysis

Package	Purpose
tidyverse, dplyr, tidyr	Data manipulation and visualization
mice	Missing Values & Pattern Plots
summarytools	Summaries and distributions
vtable	Variable tables
outliers	Outlier detection
ggplot2	Advanced graphics
patchwork	Combine plots
ggpubr	Publication plots
GGally	Pair plots
car	ANOVA
pROC	ROC curves
MLmetrics	Classification metrics
rpart, rpart.plot	Decision Trees
randomForest	Random Forests
e1071	SVM
class	kNN
naivebayes	Naive Bayes
cluster	Clustering algorithms kMeans
dbscan	DBSCAN
caret	Machine Learning
magrittr	Pipe operator
factoextra	Clustering visualization
gt	Tables
gridExtra	Grid Searches

Bibliography

References

1. Austin, P.C., White, I.R., Lee, D.S., van Buuren, S.: Missing data in clinical research: A tutorial on multiple imputation. *Canadian Journal of Cardiology* **37**(9), 1322–1331 (2021). <https://doi.org/https://doi.org/10.1016/j.cjca.2020.11.010>, <https://www.sciencedirect.com/science/article/pii/S0828282X20311119>
2. Bin Rakhis, Sr, S.A., AlDuwayhis, N.M., Aleid, N., AlBarrak, A.N., Aloraini, A.A.: Glycemic control for type 2 diabetes mellitus patients: A systematic review. *Cureus* **14**(6), e26180 (Jun 2022)
3. Blumenreich, M.S.: The white blood cell and differential count. In: *Clinical Methods: The History, Physical, and Laboratory Examinations*. Butterworths, Boston (1990)
4. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: *Classification And Regression Trees*. Routledge (Oct 2017)
5. Guido, S., Mueller, A.C.: *Introduction to machine learning with python*. O'Reilly Media, Sebastopol, CA (Oct 2016)
6. Hastie, T., Tibshirani, R., Friedman, J.: *The Elements of Statistical Learning*. Springer Series in Statistics, Springer New York Inc., New York, NY, USA (2001)
7. Melo, F.: Area under the ROC Curve, pp. 38–39. Springer New York, New York, NY (2013). https://doi.org/10.1007/978-1-4419-9863-7_209, https://doi.org/10.1007/978-1-4419-9863-7_209
8. Murtagh, F., Legendre, P.: Ward's hierarchical clustering method: Clustering criterion and agglomerative algorithm (11 2011)
9. R Core Team: *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria (2022), <https://www.R-project.org/>
10. Ros, F., Guillaume, S., Riad, R., El Hajji, M.: Detection of natural clusters via s-dbscan a self-tuning version of dbscan. *Knowledge-Based Systems* **241**, 108288 (2022). <https://doi.org/https://doi.org/10.1016/j.knosys.2022.108288>, <https://www.sciencedirect.com/science/article/pii/S0950705122000946>
11. Tukey, J.W.: *Exploratory Data Analysis*. Addison-Wesley (1977)