Comparison of machine learning algorithms for mortality prediction in intensive care patients on multi-center critical care databases

Thanakron Na Pattalung¹, Sitthichok Chaichulee¹,#
¹ Institute of Biomedical Engineering, Department of Biomedical Sciences and Biomedical Engineering, Faculty of Medicine, Prince of Songkla University, Songkhla 90110, Thailand.

E-mail address: sitthichok.c@psu.ac.th

Abstract. Current scoring systems for mortality prediction in intensive care patients are usually applied once after 24 hours of admission, as all parameters needed for scoring are not yet available. In addition, several parameters are dynamic and may change according to patient conditions. It is hypothesized that mortality prediction should be made at the earliest when relevant information becomes available and continuously during patient stay. This study focuses on the development of algorithms for mortality prediction from vital signs and laboratory results based on the data from three recent critical care databases, i.e. the eICU collaborative research database, the Multiparameter Intelligent Monitoring in Intensive Care III (MIMIC-III) database, and the MIMIC-IV database. We employed logistic regression, k-nearest neighbours, neural networks and tree-based classifiers for such problem. Our models had the area under the receiver operating characteristic curve (AUROC) ranging from 0.67 – 0.95. Reliable mortality prediction can be made as early as the first 4 hours after ICU admission. We provided comprehensive analysis on different time frames used for prediction, models trained with top attributes, models trained with data combination, and missing values. Our results provide guidelines and benchmarks for the development of such algorithm in local narratives.

1. Introduction

Patients are admitted into the intensive care unit (ICU) generally following severe illness, accidents, or surgery. They require special medical attention and are at high risk of clinical deterioration and death. The patient’s risk of death can be determined using demographic information, the severity of patient condition, vital sign measurements, and laboratory results [1]–[4]. Mortality prediction plays an important role not only for identifying high-risk patients and prioritizing patient care but also for administrative purposes, such as evaluating the performance of ICUs across health systems [5].

Conventional scoring systems for mortality prediction usually applied once after 24 hours of ICU admission as they require the aggregation of various clinical parameters in which many measurements are not yet available during the first hours of admission. The availability of clinical data stored in the modern hospital information system (HIS) allows the development of modern predictive tools for estimating patient mortality using machine learning. The tools could leverage the use of data for identifying patterns or relationship between clinical data and patient outcomes. They could make possible the early prediction of patient mortality that could benefit the effectiveness of patient care. Recently, the Massachusetts Institute of Technology (MIT) Laboratory for Computational Physiology
released several public critical care databases [6]–[8], promoting the development of machine learning algorithms on high-volume datasets. The algorithms and techniques developed on large public datasets could be used as a blueprint for developing in-house decision support systems.

This study focuses on the comparison of different machine learning algorithms for mortality prediction on three different critical care databases, namely, the eICU collaborative research database [6], the Multiparameter Intelligent Monitoring in Intensive Care III (MIMIC-III) database [7], and the MIMIC-IV database [8]. To this end, our contributions are three-fold. Firstly, we present detailed benchmarking results using the data obtained from the first 4, 8, 12, 16, 20, and 24 hours following ICU admission. We show that the data obtained during the first 4 hours are enough to make the prediction of patient mortality with clinically acceptable performance. Secondly, we compare the availability of each clinical variables in different time frames across different databases. We found that the high frequency of recorded vital signs in the eICU database could contribute to the higher performance of such algorithms developed on the eICU database, compared to those developed on the MIMIC-III and MIMIC-IV databases. The frequency of recorded vital signs in the HIS, hence, may be one of the key contributors for the improvement of the algorithms. Finally, we employ the same preprocessing pipeline for all the databases in order to standardize the data to be in the same format prior to comparison. We, therefore, present benchmarking results, separately for each database, on the models trained with all the data combined. This is different from other studies.

Our paper is organized as follows. Section 2 presents related work regarding mortality prediction using both conventional scoring systems and machine learning algorithms. Section 3 describes the critical care databases used in our study. Section 4 explains methods for training and evaluating algorithms. Section 5 details the comparison of each algorithm of each database. Section 6 discusses the results and provides suggestions for the development of algorithms in local narratives. Section 7 concludes the contributions made by this study.

2. Related work
Predicting patient’s clinical outcomes is an important yet challenging topic for research in critical care. Studies have shown that abnormalities in physiological observations are present in patients prior to their major clinical events such as infection, congestive heart failure, cardiac arrest and death. Eighty-four percent of patients had documented signs of physiological abnormalities within eight hours prior physiological derangement events [9]. Even though abnormalities in biochemistry results were shown to be inconsistent, abnormalities in vital signs were consistently observed. Several clinical scoring systems have been introduced to the ICU for predicting patient’s risk of dying based on studying aggregated data from large cohorts of patients. In this section, we reviewed related work regarding mortality prediction: conventional scoring systems, machine learning algorithms, and challenges in processing critical care data.

2.1. Conventional scoring systems for mortality prediction
Various scoring systems have been introduced for mortality prediction in the ICU, for example: the Acute Physiology and Chronic Health Evaluation (APACHE) scores [1], [2]; the Simplified Acute Physiology Score (SAPS) [4]; and the Sequential Organ Failure Assessment (SOFA) score [3]. These clinical scoring systems are mostly based on patient demographics, physiological and biochemistry variables, obtained during the first 24 hours of ICU admission. They relied on a panel of clinical and statistical experts while developing such scoring systems. Each score has its specific purpose and been seen as complementary rather than mutually exclusive [5]. The APACHE [1] and SAPS [4] scores, for example, assess the severity of illness which has an impact on patient’s mortality, while the SOFA score [3] provides an assessment of organ derangements. They obtained, in their original articles, the areas under the receiver operator characteristic curves (AUROC) between 0.68 and 0.88 [1]–[4].
2.2. Machine learning techniques for mortality prediction
Information stored in the HIS can be utilized through various techniques to observe the trends of disease progression in order to plan for prevention and intervention and to increase the effectiveness of treatment [10]. Machine learning is a tool to find a relationship between input and output data. Many machine learning algorithms have been recently developed for predicting patient mortality following ICU admission based on large public critical care databases, such as MIMIC-II/III databases [7], [11]. The availability of data in the HIS together with algorithms developed could make possible the automated and real-time prediction of patient mortality in the ICU.

Recently, Awad et al. [12] proposed the Early Mortality Prediction for Intensive Care Unit (EMICU) patients, developed on the MIMIC II database [11], in which patient mortality can be predicted as early as 6 hours after ICU admission with an AUROC of 0.90 using the Random Forests algorithm [13]. Harutyunyan et al. [14] benchmarked multi-task neural networks algorithms for predicting clinical outcomes on the MIMIC III database [7] and obtained an AUROC of 0.87 for mortality prediction using their multi-task long short-term memory algorithm (LSTM). Purushotham et al. [15] benchmarked multiple algorithms for mortality prediction on the MIMIC III database [7] and obtained an AUROC of 0.94 on 24-hour data following ICU admission using the multi-modal deep Boltzmann machines algorithm by treating temporal features and non-temporal features separately. El-Rashidy et al. [16] proposed an ensemble stacking model for early mortality prediction with an AUROC of 0.93 for the data obtained during the first 6 hours after ICU admission.

Most studies demonstrated their algorithms performed better than the traditional scoring systems used in the ICU, mainly on the MIMIC III database [7].

2.3. Techniques for processing critical care data
Many challenges are associated with the processing of clinical care data such as missing values, periodicity, variable selection, standardization, and class imbalance. Recently, Wang et al. [17] proposed the MIMIC-Extract framework for preparing data from the MIMIC-III database [7] for developing machine learning algorithms. They also released clinical taxonomy which could help data aggregation and standardization. Tang et al. [18] proposed the FIDDLE framework as a data preprocessing pipeline, evaluating on the eICU [6] and MIMIC-III [7] databases. This could further help the processing and standardization of data across different datasets possible. These frameworks could foster the development of machine learning algorithms on critical care data. Currently, at the time of manuscript preparation, no framework has been specifically designed for preprocessing the MIMIC-IV database [8].

3. Datasets
This study involves three public critical care databases: the eICU Collaborative Research Database [6], the MIMIC-III Critical Care Database [7], and the MIMIC-IV Critical Care Database [8].

3.1. eICU Collaborative Research Database v2.0
The eICU Collaborative Research Database [6], released in 2018, is a public critical care database provided by the MIT Laboratory for Computational Physiology in partnership with Philips Healthcare. The eICU contains deidentified data from over 200,000 ICU admissions from 2014 to 2015 monitored by eICU Programs throughout the United States. The database includes vital sign measurements, biochemistry measurements, patient diagnosis, treatment information, care plan documents, severity-of-illness measures, etc.

3.2. MIMIC-III Critical Care Database v1.4
The MIMIC-III Critical Care Database [7], released in 2016, is a public critical care database obtained from patients admitted to ICUs at the Beth Israel Deaconess Medical Center (BIDMC) in Boston, Massachusetts between 2001 and 2012. It contains deidentified data from over 38,597 distinct adult...
patients with 49,785 hospital admissions. The database has patient demographics, vital sign measurements, laboratory tests, doctor notes, clinical events, etc.

3.3. MIMIC-IV Critical Care Database v0.4
The MIMIC-IV Critical Care Database [8], released in 2020, a public critical care database covering over 200,000 emergency department (ED) stays and 60,000 ICU stays obtained from hospitalized patients from BIDMC between 2008 and 2019. The database contains the clinical data prior to ICU admission as 65% of patients admitting to the ICU was first seen in the ED. For critical care data, the MIMIC-IV database has similar structure to the MIMIC-III database.

4. Methods
This section describes the steps we employed for cohort selection, variable selection, data cleansing and standardization, data aggregation, data imputation, training machine learning algorithms and evaluating such algorithms. The first part of these steps was designed to transform the data to be in the same format such that the comparison of different algorithms and the stacking of data from different databases were made possible.

4.1. Cohort selection
We applied the same criteria to all the databases for building our study cohorts. We included adult patients who were firstly admitted to the ICU for more than 24 hours into our study. We excluded patients without recorded vital signs or laboratory measurements. For the eICU database, we used the unitdischargestatus variable in the patients table as an indicator whether patients were dead or survived their first ICU stay. For the MIMIC-III and MIMIC-IV databases, the death of patients was indicated in the deathtime and hospital_expire_flag variables in the admission table. Our criteria for cohort selection is consistent with the studies by [12], [16]. Patients who were dead during their first ICU stay were set as our experimental (or positive) group. Table 1 summarizes the demographics of our study cohorts. Patients who were not dead during their hospital stays were set as our control (or negative) group. We performed random under-sampling for the control group to have its demographics match that of the experiment group during the evaluation of machine learning algorithms.

| Demographic | Dataset | eICU | MIMIC III | MIMIC IV |
|-------------|---------|------|-----------|----------|
| Number of patients | eICU | 5,519 | 3,548 | 4,103 |
| Gender | | | | |
| Male | 3,053 | 1,881 | 2,202 |
| Female | 2,464 | 1,667 | 1,901 |
| Unknown | 2 | 0 | 0 |
| Ethnicity | | | | |
| Caucasian | 4,300 | 2,414 | 2,474 |
| African American | 577 | 195 | 300 |
| Hispanic | 167 | 70 | 101 |
| Asian | 106 | 79 | 124 |
| Others/Unknown | 369 | 790 | 1,104 |

* Our study cohorts contain adult patients who were admitted to the ICU for more than 24 hours and were dead during their first ICU stay.
4.2. Variable selection

Vital sign and laboratory variables selected for the development of algorithms were based on the review of conventional clinical scoring systems [1]–[4] that are relevant to mortality prediction and the consultation with clinical care experts. We chose only the variables presenting in all the databases. Table 2 lists all vital sign variables (7 variables) and laboratory variables (19 variables).

Table 2. Vital sign and laboratory variables obtained from the eICU, MIMIC III, and MIMIC IV databases for benchmarking machine learning algorithms

| Variables                      | Dataset * | Measurement unit        |
|--------------------------------|-----------|-------------------------|
|                               | eICU      | MIMIC-III | MIMIC-IV   |
| Vital signs                    |           |           |            |
| Heart Rate                     | V         | C         | C          |
| Respiratory Rate               | V         | C         | C          |
| Peripheral O2 saturation       | V         | C         | C          |
| Blood Pressure mean            | V         | C         | C          |
| Blood Pressure systolic        | V         | C         | C          |
| Blood Pressure diastolic       | V         | C         | C          |
| Temperature                    | V         | C         | C          |
| Laboratory results             |           |           |            |
| Bilirubin                      | L         | L/C       | L/C        |
| Bicarbonate (HCO3)             | L         | L/C       | L/C        |
| Calcium                        | L         | L/C       | L/C        |
| Chloride                       | L         | L         | L          |
| Creatinine                     | L         | L         | L          |
| Glucose                        | L         | L         | L          |
| Hemoglobin                     | L         | L         | L          |
| Hematocrit                     | L         | L         | L          |
| O2 Saturation                  | L         | L         | L          |
| Inspired O2 Fraction (FiO2)    | L         | C         | C          |
| Lactate                        | L         | L/C       | L/C        |
| Partial pressure of CO2        | L         | L         | L          |
| Partial Pressure of O2         | L         | L         | L          |
| pH                             | L         | L         | L          |
| Potassium                      | L         | L         | L          |
| Sodium                         | L         | L         | L          |
| Platelet Count                 | L         | L         | L          |
| Red Blood Cells                | L         | L         | L          |
| White Blood Cells              | L         | L         | L          |

* ‘L’ denotes the variable(s) extracted from the labevents table
‘C’ denotes the variable(s) extracted from the chartevents table
‘V’ denotes the variable(s) extracted from the vitalperiodic table

4.3. Data standardization

In the databases, each measurement was associated with a unique type identifier defined by the original EMR system. Glucose, for example, may be recorded under different names such as GLUCOSE, Glucose, BloodGlucose, or Fingerstick Glucose. As provided by [17], we grouped the semantically equivalent measurements together using the clinical taxonomies provided by [18] and [17]. This could reduce the number of missing observations. Since our experiments were performed on different databases, we
standardize measurements by performing unit conversion for both intra-database and inter-database. We used the units employed in the MIMIC-IV database (released in 2020) as the standard. In order to handle outliers, we used clinical validity ranges provided by [18], [17], and [19]. Any observation falling outside its validity range was removed.

### 4.4. Data aggregation
Each measurement was associated with a fine-grained timestamp. Vital sign measurements were mostly frequent, with an average of >1 time per hour in the eICU and MIMIC III databases [18], while laboratory measurements were infrequent. In order to fit the data to machine learning models, we aggregated the measurements hourly for frequently recorded vital signs and every four hours for laboratory measurements. For aggregation, we applied the following summary statistics functions: minimum and maximum. These were similar to [17] and were in consistent with the recommendations made by [18].

### 4.5. Data imputation
For each variable, missing data were flagged (1 if a data point was missing and 0 otherwise, for each timestep) then first imputed by forward filled from their previous values. If the variable was never observed in this patient, its values were set to -1 for all timesteps.

### 4.6. Machine learning algorithms
We evaluated six different machine learning algorithms on the eICU, MIMIC-III, and MIMIC-IV databases. Note that we performed standardization, as detailed above, such that the data obtained from each dataset has the same set of variables. Our data were transformed into a binary classification problem in which the event of patients died during their ICU stay or survived was set as a target. For each algorithm, we applied a range of hyperparameter configurations in order to obtain a model with the best hyperparameter setting. The overview and implementation details of each algorithm were as followed:

**Logistic regression** – Logistic regression [20] was used to statistically model the probability of a certain event (survived/dead) using a binary logistic function. Our data were fitted into a linear logistic regression model. We used L2 regularization and varied a regularization parameter (lambda) between 0 and 1 with an increment of 0.25.

**K-nearest neighbors** – K-nearest neighbors [20] utilizes local spatial information to estimate the outcome of a new sample. For example, the algorithm with 100-nearest neighbors employs the hundred closest samples, based on a distance metric measured in multidimensional space, to predict the outcome of the new sample. We varied the number of nearest neighbors between 100 and 900 with an increment of 200.

**Neural networks** – Neural networks [20] utilizes a non-linear computation of input variables to predict an outcome of the event of interest. Inputs were passed into hidden layer(s) in which each neuron in that layer performs a linear combination of all neurons in the previous layer and applies a non-linear transform through an activation function. The resulting values were then passed to the next hidden layer or the output layer. We employed a feed-forward multilayer perceptron model. We varied the number of hidden layers from 1 to 3 and the number of neurons in each layer from 32 to 128 with an increment of 32.

**Bagging (Bootstrap Aggregation)** – Bagging [20] is an ensemble method for improving the robustness of weak classifiers. It could be implemented with any machine learning algorithms. We employ Bagging with decision tree in this work. Decision tree [20] involves growing and pruning a tree model by determining the best variable to split and the best splitting value from the training data. A single decision tree, however, often has suboptimal predictive performance. Bagging has a collection of many decision trees in which each tree is trained by taking random samples with all features from the full dataset with replacement. Final decision was made by finding a consensus from all decision trees. We varied the number of decision trees from 100 to 500 with an increment of 200.
**Random Forests** – Random Forests [13] is similar to Bagging with decision trees; however, only a subset of features was selected at random for training each decision tree. We varied the number of decision trees from 100 to 500 with an increment of 200 and a choice of the number of features for each tree (the square root of the number of all features or the common logarithm of the number of all features).

**Gradient Boosting** – Gradient Boosting [20] involves consecutive training of weak decision trees on the residual of previous models, gradually correcting errors it made. We varied the maximum depth of trees (3, 7, 13, 49 and 98), the number of boosting stages (200, 400, and 600), and the learning rate (0.025, 0.050, 0.100).

All experiments were implemented using the Python programming language version 3.8.5 (Python Software Foundation, https://www.python.org/), the pandas data analysis library [21], and the scikit-learn machine learning library version 0.23.2 [22] on a computer with 64 processing cores and 256 gigabytes of physical memory.

4.7. Evaluation procedure
The development of machine learning algorithms requires holding out data for both hyperparameter tuning and performance evaluation. The use of traditional k-fold cross validation, in which the dataset is partitioned into k distinct parts and one part is iteratively used for validation and the remaining parts are then used for training, may cost information leakage leading to optimistically biased results [23] as it employs the same set of data for both optimizing hyperparameters and evaluating performance.

We employed nested cross validation, in which two rounds of cross validation was performed: one for finding the best hyperparameters and the other for testing the resultant model on a hold-out set. Nested cross validation technique was shown to produce almost unbiased estimation of performance with low variance [24]; hence, it is a recommended technique for algorithm comparisons for small and moderately-sized datasets.

In our experiments, nested cross validation with 10 outer folds and 10 inner folds was used. A dataset was first randomly partitioned into 10 parts for outer cross validation. In each outer round, one part was held out for testing and the remaining parts were used for developing a model. For model development, the remaining parts were then further randomly partitioned into 10 parts for inner cross validation. In each inner round, one part was held out for model validation and the remaining parts were used for training a model. Multiple hyperparameter combinations were evaluated in inner cross validation, then a model with the best hyperparameter was chosen for testing with the hold-out outer test set. Results were then aggregated over different outer rounds.

The receiver operating characteristic (ROC) curves were used to provide insights on the results from different algorithms. The area under the ROC curves (AUROC) was used to provide a summary of the ROC curve.

5. Results
Table 3 shows the comparison of machine learning algorithms on critical care different databases. For each experiment, the AUROC was averaged from all outer cross-validation folds in our nested cross-validation scheme and its standard deviation was also computed. Gradient boosting yielded the highest AUROCs for all three databases, followed by Random Forests. The data obtained during the first 4 hours following ICU admission had acceptable discriminative power for predicting mortality with the AUROCs of 0.89, 0.83 and 0.78 for the gradient boosting models trained on the eICU, MIMIC-III and MIMIC-IV databases, respectively. For each algorithm and each time frame, the models trained and evaluated on the eICU database had the highest AUROCs, compared to those trained on the MIMIC-III or MIMIC-IV databases. K-nearest neighbors yielded the lowest AUROCs in all cases. For all experiments, standard deviations were very low, generally less than 0.02. Figure 1 shows ROC curves comparing different algorithms on different databases for the data obtained from the first four hours.

Table 4 describes the comparison of performances of gradient boosting models for mortality prediction trained on different attributes. The models trained with the top 10 clinical variables, ranking by their feature importance, had similar AUROCs compared to those trained with all clinical variables.
The gradient boosting models trained with only vital sign variables or with only laboratory variables yielded, as expected, lower AUROCs compared to those trained with both groups of clinical variables; nevertheless, these models still have acceptable discriminating power for mortality prediction.

| Table 3. Performance of different machine learning models for mortality prediction |
|-----------------------------------------------|
| **Algorithm** | **Dataset** | **AUROC (mean ± standard deviation)** |
| | | **4 hours** | **8 hours** | **12 hours** | **16 hours** | **20 hours** | **24 hours** |
| Logistic regression | eICU | 0.83 ± 0.01 | 0.86 ± 0.01 | 0.88 ± 0.01 | 0.89 ± 0.01 | 0.90 ± 0.01 | 0.90 ± 0.01 |
| | MIMIC III | 0.79 ± 0.01 | 0.82 ± 0.01 | 0.83 ± 0.01 | 0.83 ± 0.01 | 0.84 ± 0.01 | 0.85 ± 0.01 |
| | MIMIC IV | 0.75 ± 0.02 | 0.73 ± 0.02 | 0.74 ± 0.01 | 0.75 ± 0.02 | 0.76 ± 0.02 | 0.79 ± 0.01 |
| K-nearest neighbors | eICU | 0.78 ± 0.01 | 0.80 ± 0.01 | 0.80 ± 0.01 | 0.81 ± 0.01 | 0.81 ± 0.01 | 0.81 ± 0.01 |
| | MIMIC III | 0.76 ± 0.01 | 0.77 ± 0.02 | 0.78 ± 0.01 | 0.78 ± 0.01 | 0.79 ± 0.01 | 0.79 ± 0.01 |
| | MIMIC IV | 0.68 ± 0.01 | 0.67 ± 0.01 | 0.68 ± 0.02 | 0.68 ± 0.02 | 0.69 ± 0.02 | 0.69 ± 0.02 |
| Neural networks | eICU | 0.85 ± 0.01 | 0.86 ± 0.01 | 0.86 ± 0.01 | 0.87 ± 0.01 | 0.89 ± 0.01 | 0.90 ± 0.01 |
| | MIMIC III | 0.79 ± 0.01 | 0.80 ± 0.01 | 0.80 ± 0.02 | 0.82 ± 0.01 | 0.82 ± 0.02 | 0.83 ± 0.02 |
| | MIMIC IV | 0.70 ± 0.01 | 0.70 ± 0.01 | 0.70 ± 0.01 | 0.72 ± 0.02 | 0.73 ± 0.01 | 0.74 ± 0.01 |
| Random Forests | eICU | 0.88 ± 0.01 | 0.89 ± 0.01 | 0.90 ± 0.01 | 0.92 ± 0.01 | 0.92 ± 0.01 | 0.93 ± 0.01 |
| | MIMIC III | 0.82 ± 0.01 | 0.85 ± 0.01 | 0.86 ± 0.01 | 0.87 ± 0.01 | 0.87 ± 0.01 | 0.88 ± 0.01 |
| | MIMIC IV | 0.77 ± 0.01 | 0.78 ± 0.02 | 0.79 ± 0.01 | 0.80 ± 0.01 | 0.81 ± 0.01 | 0.82 ± 0.01 |
| Gradient boosting | eICU | **0.89 ± 0.01** | **0.91 ± 0.01** | **0.92 ± 0.01** | **0.93 ± 0.01** | **0.94 ± 0.01** | **0.94 ± 0.01** |
| | MIMIC III | **0.83 ± 0.01** | **0.85 ± 0.01** | **0.86 ± 0.01** | **0.87 ± 0.01** | **0.88 ± 0.01** | **0.88 ± 0.01** |
| | MIMIC IV | **0.78 ± 0.01** | **0.80 ± 0.01** | **0.80 ± 0.01** | **0.82 ± 0.01** | **0.83 ± 0.01** | **0.83 ± 0.01** |
| Bagging | eICU | 0.85 ± 0.01 | 0.86 ± 0.01 | 0.88 ± 0.01 | 0.89 ± 0.01 | 0.90 ± 0.01 | 0.90 ± 0.01 |
| | MIMIC III | 0.80 ± 0.01 | 0.82 ± 0.01 | 0.83 ± 0.01 | 0.84 ± 0.01 | 0.85 ± 0.01 | 0.85 ± 0.01 |
| | MIMIC IV | 0.73 ± 0.01 | 0.74 ± 0.01 | 0.75 ± 0.01 | 0.76 ± 0.01 | 0.77 ± 0.01 | 0.78 ± 0.01 |

**Figure 1.** ROC curves comparing different algorithms on different databases for the data obtained from the first four hours of ICU admission.

Table 5 details the performance of gradient boosting models for mortality prediction when trained with the data stacked from all the databases combined. Results reported separately for each database. Since the data prepared from all databases were in the same format, we combined all the databases by concatenating them together prior to evaluation. Improvements were seen on the predictions made on the MIMIC-IV database, while similar performances were seen on the predictions made on the other databases.
6. Discussion

Here, we discuss our results described in Section 3, provide comparisons of ours with other related work, give insights on the availability of data during the first 24 hours following ICU admission, provide limitations of our study, and give suggestions for further research and development.

6.1. Comparison across different algorithms

Tree-based classifiers (Random Forests, Gradient Boosting and Bagging) had similar trends of AUROCs compared to other algorithms, while Gradient Boosting achieved the highest AUROCs for all time frames. The k-nearest neighbors models had the lowest AUROCs for all time frames. This might be because the complexity of ICU data could not be well handled with this algorithm.

Our results are consistent with those of [16] which reported the effectiveness of ensemble tree models for mortality prediction on the first 24-hour time window. On the MIMIC-III database, Our results are slightly higher than those of [17], [18], [25], [26] for most algorithms while having similar trends. These might be because we use different variable sets and different evaluation scheme.

Note that we did not give a direct comparison of our results with traditional scoring systems as further assumptions have to be made in order to produce such scores. According to the APACHE-II, SAPS-I, SAPS-II and SOFA scores computed by [15], [26], our machine learning models attained higher AUROCs than conventional clinical scoring systems for mortality prediction.

6.2. Data availability

We calculated an average number of vital sign and laboratory measurements made following ICU admission for each patient, as illustrated in Figure 2. The volume of vital sign measurements in the eICU database was six to ten folds compared to those made in the MIMIC-III and MIMIC-IV databases. The number of laboratory measurements were similar across all databases. The increment of data has a linear trend.
Table 6: Number of measurements made for each clinical variable, counting for each time period following ICU admission for each critical care database.

| Variable                  | 0 - 4 hours | 4 - 8 hours | 8 - 12 hours | 12 - 16 hours | 16 - 20 hours | 20 - 24 hours |
|---------------------------|-------------|-------------|--------------|---------------|---------------|---------------|
| Vital signs               |             |             |              |               |               |               |
| Heart Rate                | 43.7        | 6.3         | 7.1          | 58.8          | 7.1           | 7.0           |
| Respiratory Rate          | 40.2        | 6.1         | 7.0          | 53.8          | 6.9           | 6.9           |
| Peripheral O2 saturation  | 39.9        | 2.2         | 6.6          | 54.1          | 2.5           | 6.7           |
| Blood Pressure mean       | 10.4        | 0.5         | 1.7          | 18.7          | 0.9           | 2.4           |
| Blood Pressure systolic   | 10.2        | 0.5         | 1.7          | 18.7          | 0.9           | 2.4           |
| Blood Pressure diastolic  | 10.2        | 0.5         | 1.7          | 18.7          | 0.9           | 2.4           |
| Temperature               | 5.8         | 0.1         | 0.2          | 9.0           | 0.1           | 0.3           |
| Laboratory results        |             |             |              |               |               |               |
| Bilirubin                 | 0.4         | 0.4         | 0.6          | 1.2           | 1.0           | 1.6           |
| Bicarbonate (HCO3)        | 1.1         | 0.8         | 1.4          | 1.8           | 1.4           | 1.2           |
| Calcium                   | 0.5         | 0.6         | 0.7          | 1.2           | 1.5           | 1.8           |
| Chloride                  | 0.5         | 0.6         | 0.7          | 0.9           | 1.0           | 1.2           |
| Creatinine                | 0.5         | 0.6         | 0.7          | 0.9           | 1.0           | 1.2           |
| Glucose                   | 0.5         | 0.8         | 0.9          | 1.0           | 1.4           | 1.6           |
| Hemoglobin                | 0.5         | 0.7         | 0.9          | 1.2           | 1.4           | 1.4           |
| Hematocrit                | 0.5         | 0.6         | 0.7          | 0.9           | 1.1           | 1.2           |
| O2 Saturation             | 0.5         | 0.3         | 0.2          | 0.8           | 0.5           | 0.4           |
| Inspired O2 Fraction      | 0.7         | 0.4         | 1.3          | 1.0           | 0.8           | 2.4           |
| Lactate                   | 0.4         | 0.5         | 0.3          | 0.7           | 0.7           | 1.3           |
| Partial pressure of CO2   | 0.6         | 0.8         | 0.8          | 1.0           | 1.5           | 1.5           |
| Partial Pressure of O2    | 0.6         | 0.8         | 0.8          | 1.0           | 1.5           | 1.5           |
| pH                        | 0.6         | 0.9         | 0.9          | 1.0           | 1.6           | 1.6           |
| Potassium                 | 0.6         | 0.6         | 0.7          | 1.1           | 1.0           | 1.2           |
| Sodium                    | 0.6         | 0.6         | 0.7          | 1.1           | 1.0           | 1.2           |
| Platelet Count            | 0.4         | 0.5         | 0.7          | 0.7           | 1.0           | 1.1           |
| Red Blood Cells           | 0.4         | 0.5         | 0.7          | 0.7           | 0.9           | 1.1           |
| White Blood Cells         | 0.4         | 0.5         | 0.7          | 0.7           | 0.9           | 1.1           |

Table 6 details the number of measurements made for each clinical variable, counting for each time period following ICU admission for each critical care database. The eICU database has more frequent vital sign measurements. Temperature measurements was not frequently recorded in the MIMIC-III and MIMIC-IV databases. The number of laboratory measurements was similar across all databases. During

![Figure 2](a) (b)

Figure 2. Average number of measurements made after ICU admission for (a) recorded vital signs and (b) laboratory results for each critical care database.
the first four hours of ICU admission, we could expect half of laboratory measurements were made. The measurements were likely to complete during the first eight hour of ICU admission. The different in frequencies of recorded vital signs and laboratory measurements leading to us choosing different time windows (every hour for vital sign measurements and every four hours for laboratory measurements) for data aggregation.

6.3. Comparison across databases
The machine learning models evaluated on the eICU database had the highest AUROCs, followed by those evaluated on the MIMIC-III and MIMIC-IV databases, respectively. This might be because the high frequency of data collected in the eICU database providing more discriminative features. Even though the MIMIC-IV database is a successor to the MIMIC-III database, both databases contain similar yet different sets of data: the MIMIC-III database contains data from 2001-2012; and the MIMIC-IV database contains data from 2008 to 2019, both from BIDMC. The differences in their results could be expected. For the models trained on the combination of data from all databases, small improvements in the AUROCs were seen on predictions made on the MIMIC-IV database. The amount of data used for training machine learning algorithms might contribute to these improvements. Note that at the time of manuscript preparation, the MIMIC-IV was still in development.

6.4. Limitations
Our study was subject to several limitations. Firstly, we did not attempt to model temporal relationship of continuous variables. All experiments were performed only on traditional machine learning algorithms in which clinical features were aggregated using summary statistics and were treated the same. This was unlike the study by Purushotham et al. [15] in which non-temporal features (e.g. laboratory tests) and temporal features (e.g. vital signs) were treated separately using recurrent neural networks. Their study yielded an AUROC of 0.94 on 24-hour data following ICU admission, compared to ours whose AUROC was 0.88 for the same period. A room of improvements could still be achieved.

Secondly, we did not perform any experiments on data imputation apart from using carry-forward imputation and a missing value flag. We did not observe any improvements in our initial experiments in which missing values were imputed as in [17], [26] with global mean, median or normal values. This was also consistent by what reported in [18].

Finally, we performed under-sampling for patients in the control group (survived during hospital stay). This was done as our computation resources were limited and we did not see much improvement over the use of oversampling techniques, reported in [26].

6.5. Suggestions
Conventional clinical scoring systems have been developed and employed in the clinical practice for more than two decades. Currently, modern HIS has been a hub of extensive clinical data collections including electronic medical notes, laboratory results, physiological values excerpted from bedside monitoring systems, information regarding fluids delivered to patients via modern infusion pumps, or data derived from mechanical ventilators. These clinical data could be used for the development of clinical support systems. These collections of data are, however, different from hospital to hospital, depending on their context, culture and available resources.

For the automation of clinical scoring processes, one could develop and validate algorithms using the data gathered from the existing HIS together with the data obtained from public clinical databases. However, the utilization of different data requires careful considerations on feature transformations and the algorithms should be less dependent on the features that are hard to obtain or not currently available from the existing HIS.

We found that most tree-based algorithms, such as Random Forests and Gradient Boosting, can work out of the box with slight tweaks on hyperparameter optimization. Carry-forward imputation with a missing value indication could be utilized in the case that some measurements are not yet available. Several clinical outcome prediction algorithms could be combined and formed a multitask algorithm as
Techniques for modelling temporal features explicitly such as recurrent neural networks could be employed; however, special attentions may need for data transformations as clinical measurements across different systems or databases are usually collected at different frequencies.

7. Conclusion

We performed comprehensive analysis of different machine learning algorithms for mortality prediction on three critical care databases: the eICU, MIMIC-III and MIMIC-IV databases. We achieved similar performances for tree-based classifiers, whilst Gradient Boosting attained the highest AUROC for all the databases. Clinical parameters obtained during the first 4 hours of ICU admission are enough for predicting patient mortality with clinically acceptable performance. The models trained with the top ten clinical features, raking by their importance, yielded similar performances compared to those trained with all clinical variables. The models trained with the data combined from all the databases yielded slight improvements in predictive performance. Clinical features needed for scoring are mostly available after 8 hours of ICU admission. Vital sign measurements in the eICU database are six to ten times more frequent than those in the MIMIC-III and MIMIC-IV databases. These likely to contribute to the higher performance of the algorithms trained on the eICU database, compared to those trained on the other databases. Our study provides guidelines and benchmarks for developing algorithms for the prediction of clinical outcomes in local narratives.

References

[1] W. A. Knaus, E. A. Draper, D. P. Wagner, and J. E. Zimmerman, “APACHE II: a severity of disease classification system,” Crit. Care Med., vol. 13, no. 10, pp. 818–29, Oct. 1985.
[2] W. A. Knaus et al., “The APACHE III prognostic system: Risk prediction of hospital mortality for critically III hospitalized adults,” Chest, 1991.
[3] J. L. Vincent et al., “The SOFA (Sepsis-related Organ Failure Assessment) score to describe organ dysfunction/failure,” Intensive Care Med., 1996.
[4] J. R. Gall, S. Lemeshow, and F. Saulnier, “A New Simplified Acute Physiology Score (SAPS II) Based on a European/North American Multicenter Study,” JAMA J. Am. Med. Assoc., 1993.
[5] J. Lee, J. A. Dubin, and D. M. Maslove, “Mortality Prediction in the ICU,” in Secondary Analysis of Electronic Health Records, 2016, pp. 1–427.
[6] T. J. Pollard, A. E. W. Johnson, J. D. Raffia, L. A. Celi, R. G. Mark, and O. Badawi, “The eICU Collaborative Research Database, a freely available multi-center database for critical care research,” Sci. Data, vol. 5, no. 1, p. 180178, Dec. 2018.
[7] A. E. W. Johnson et al., “MIMIC-III, a freely accessible critical care database,” Sci. Data, 2016.
[8] A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark, “MIMIC-IV (version 0.4),” PhysioNet, 2020.
[9] R. M. H. Schein, N. Hazday, M. Pená, B. H. Ruben, and C. L. Sprung, “Clinical Antecedents to In-Hospital Cardiopulmonary Arrest,” Chest, vol. 98, no. 6, pp. 1388–1392, Dec. 1990.
[10] M. A. F. Pimentel et al., “Towards a robust estimation of respiratory rate from pulse oximeters,” IEEE Trans. Biomed. Eng., vol. 64, no. 8, pp. 1914–1923, 2016.
[11] M. Saeed et al., “Multiparameter intelligent monitoring in intensive care II: A public-access intensive care unit database,” in Critical Care Medicine, 2011.
[12] A. Awad, M. Bader-El-Den, J. McNicholas, and J. Briggs, “Early hospital mortality prediction of intensive care unit patients using an ensemble learning approach,” Int. J. Med. Inform., 2017.
[13] L. Breiman, “Random Forests,” Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.
[14] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, and A. Galstyan, “Multitask learning and benchmarking with clinical time series data,” Sci. Data, vol. 6, no. 1, pp. 1–18, 2019.
[15] S. Purushotham, C. Meng, Z. Che, and Y. Liu, “Benchmarking deep learning models on large healthcare datasets,” *J. Biomed. Inform.*, vol. 83, no. March, pp. 112–134, 2018.

[16] N. El-Rashidy, S. El-Sappagh, T. Abuhmed, S. Abdelrazek, and H. M. El-Bakry, “Intensive Care Unit Mortality Prediction: An Improved Patient-Specific Stacking Ensemble Model,” *IEEE Access*, vol. 8, pp. 133541–133564, 2020.

[17] S. Wang, M. B. A. McDermott, G. Chauhan, M. Ghassemi, M. C. Hughes, and T. Naumann, “MIMIC-Extract,” in *Proceedings of the ACM Conference on Health, Inference, and Learning*, 2020, pp. 222–235.

[18] S. Tang, P. Davarmanesh, Y. Song, D. Koutra, M. W. Sjoding, and J. Wiens, “Democratizing EHR analyses with FIDDLE: a flexible data-driven preprocessing pipeline for structured clinical data,” *J. Am. Med. Informatics Assoc.*, 2020.

[19] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, and A. Galstyan, “Multitask learning and benchmarking with clinical time series data,” *Sci. Data*, 2019.

[20] C. M. Bishop, *Pattern recognition and machine learning*, 6th ed. New York, USA: Springer-Verlag, 2006.

[21] W. McKinney, “Data Structures for Statistical Computing in Python,” in *Proceedings of the 9th Python in Science Conference*, 2010.

[22] F. Pedregosa *et al.*, “Scikit-learn: Machine learning in Python,” *J. Mach. Learn. Res.*, 2011.

[23] S. Varma and R. Simon, “Bias in error estimation when using cross-validation for model selection,” *BMC Bioinformatics*, 2006.

[24] A. Vabalas, E. Gowen, E. Poliakoff, and A. J. Casson, “Machine learning algorithm validation with a limited sample size,” *PLoS One*, 2019.

[25] S. Purushotham, C. Meng, Z. Che, and Y. Liu, “Benchmarking deep learning models on large healthcare datasets,” *J. Biomed. Inform.*, 2018.

[26] A. Awad, M. Bader-El-Den, J. McNicholas, J. Briggs, and Y. El-Sonbaty, “Predicting hospital mortality for intensive care unit patients: Time-series analysis,” *Health Informatics J.*, 2020.

**Acknowledgement**

TNP acknowledges funding for his graduate study from Faculty of Medicine, Prince of Songkla University. He also received funding for research from Graduate School, Prince of Songkla University. This work was supported by the Health Systems Research Institute (HSRI) under grant number HSRI 63-147. This work was also supported by the Kasikornthai Foundation through the ‘PSU Fights for COVID-19’ funding. The authors gratefully acknowledge the support of computational resources from the Medical Data Center for Research and Innovation (MDCRI), Faculty of Medicine, Prince of Songkla University.