Predicting Discharge Destination of Critically Ill Patients Using Machine Learning

Zahra Shakeri Hossein Abad, *Member, IEEE*, David M. Maslove, and Joon Lee, *Senior Member, IEEE*

***Abstract*—Decision making about discharge destination for critically ill patients is a highly subjective and multidisciplinary process, heavily reliant on the ICU care team, patients and their caregivers' preferences, resource demand, staffing, and bed capacity. Timely identification of discharge disposition can be useful in care planning, and as a surrogate for functional status outcomes following critical illness. Although prior research has proposed methods to predict discharge destination in a critical care setting, they are limited in scope and in the generalizability of their findings. We proposed and implemented different machine learning architectures to determine the efficacy of the Acute Physiology and Chronic Health Evaluation (APACHE) IV score as well as the patient characteristics that comprise it to predict the discharge destination for critically ill patients within 24 hours of ICU admission. We conducted a retrospective study of ICU admissions within the eICU Collaborative Research Database (eICU-CRD) populated with de-identified clinical data from adult patients admitted to an ICU between 2014 and 2015. Machine learning models were developed to predict four discharge categories: death, home, nursing facility, and rehabilitation. These models were trained and tested on 115,248 unique ICU admissions. To mitigate class imbalance, we used synthetic minority over-sampling techniques. Hierarchical and ensemble classifiers were used to further study the impact of imbalanced testing set on the performance of our predictive models. Amongst all of the tested models, XGBoost provided the best discrimination performance with an area under the receiver operating characteristic curve of 90% (recall: 71%, F1: 70%). Our findings indicate that the variables used in the APACHE IV model for estimating patient severity of illness are better predictors of hospital discharge destination than the APACHE IV score alone. Incorporating these models into clinical decision support systems may assist patients, caregivers, and the ICU team to begin disposition planning as early as possible during the hospitalization.**

***Index Terms*—Ensemble and hierarchical learning, health informatics, machine learning, patient outcome prediction.**

Manuscript received January 30, 2020; revised April 11, 2020 and May 13, 2020; accepted May 14, 2020. Date of publication May 19, 2020; date of current version March 5, 2021. This work was supported by Postdoctoral Scholarships from the Libin Cardiovascular Institute and Cumming School of Medicine, University of Calgary, as well as a Discovery Grant from the Natural Sciences and Engineering Research Council of Canada. *(Corresponding author: Zahra Shakeri Hossein Abad.)*

Zahra Shakeri Hossein Abad is with the Data Intelligence for Health Lab, Cumming School of Medicine, University of Calgary, Calgary, AB T2N 4Z6, Canada (e-mail: zshakeri@ucalgary.ca).

David M. Maslove is with the Department of Critical Care Medicine, Queen's University, Kingston, ON K7L 3N6, Canada (e-mail: david.maslove@queensu.ca).

Joon Lee is with the Data Intelligence for Health Lab and Departments of Community Health Sciences and Cardiac Sciences, Cumming School of Medicine, University of Calgary, Calgary, AB T2N 4Z6, Canada (e-mail: joonwu.lee@ucalgary.ca).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/JBHI.2020.2995836

I. Introduction

CRITICAL care is a medical service for patients with life-threatening illnesses or injuries who require advanced organ system supports [1]. The burden of critical care is massive and increasing globally. From 2000 to 2010, the annual critical care costs in the United States doubled from $56 to $108 billion, and its proportion of Gross Domestic Product (GDP) increased by 32.1% (0.54% to 0.72%, of $10,285 to $14,964 trillion) during that decade [2]. Following discharge from the intensive care unit (ICU), patients are at risk for recurrent clinical deterioration, functional impairment, and death. They may require prolonged hospital stays, readmission to the ICU, rehabilitation, or chronic-care in a skilled nursing facility [3]. Prolonged ICU stays are stressful for both patients and their families. However, premature hospital discharge could result in the patient's deterioration, unexpected readmission, and even death [4].

Considering the highly subjective and multidisciplinary nature of decision making about discharge destination, many members of the ICU care team are involved: physicians, nurses, rehab specialists, case managers, patients, and their caregivers. This high level of variability and complexity mean that determinations about a patient's ultimate disposition cannot be made until the very end of their ICU stay. Predictive methods that can effectively predict discharge destinations within the first 24 hours after ICU admission may, therefore, be useful in terms of resource allocation, decreasing ICU length of stay, and reassurance of discharge readiness [5].

Despite these considerations, existing models to predict hospital discharge destinations are rare in critical care literature and nonexistent in using the data measured during the first 24 hours of ICU admission. Most of the studies on predicting destination after hospital discharge have mainly focused on patients with neurological, musculoskeletal, and cardiopulmonary problems in an acute care setting outside of intensive care. Agarwal *et al.* [6] used a logistic regression model to predict discharge destination for stroke rehabilitation patients and showed that age, gender, and premorbid social support were significant predictors.

In similar work, Mauthe *et al.* [7] showed that functional scales and the Functional Independence Measure (FIM) are sufficient determinants of discharge deposition for stroke patients in the acute care setting. Some other studies on brain-injured patients explored the significance of severity-related, sociobiological, and socioeconomic factors on discharge destination prediction. In this study, we take advantage of ML and develop predictive models to assist hospital care providers in making timely and accurate decisions about discharge destination. To achieve this goal and given the fact that discharge destinations are closely related to patients' severity of illness [8], our objectives were two-fold: (1) to explore whether the APACHE IV score can be used to predict hospital discharge destination for critically ill patients; and (2) to combine the components of the APACHE IV score with machine learning algorithms to develop new predictive models for discharge destinations. We chose APACHE IV score and its comprising variables for this study because of the widespread use of APACHE scores in critical care units and their demonstrated reliability and usability.

Even the relatively few studies on discharge prediction in a critical care setting are limited in scope; they exclusively focused on discharge readiness [4], [8]–[10] or they are designed to predict a specific discharge destination [11]–[13]. For example, Szubski *et al.* [11] used demographic, ICU admission, and ICU clinical data measured during the first 24 hours of ICU admission to develop a predictive algorithm for early identification of ICU patients with a high probability of discharge to a long-term acute care hospital (LTACH). They found that their predictive algorithm can accurately predict the likelihood of LTACH discharges among ICU patients. Brook *et al.* [12] investigated the relationship between Vitamin D status at ICU admission and Home/non-Home discharge destination for critically ill surgical patients. They suggested that vitamin D level may impact patient-oriented outcomes in ICU patients and it might be a modifiable risk factor for discharge destination. Compared to these studies, our proposed approach is distinct in four ways: (1) to benefit from timely identification of discharge disposition, we develop predictive models to effectively predict hospital discharge destination using the data from the first 24 hours after ICU admission, (2) in addition to traditional ML algorithms, we create a hierarchical classifier, a stacked ensemble model, and a classification-clustering ensemble (CCE) to address the problem of class imbalance, (3) to explore the informativeness of the presence/absence of predictor variables and to address the problem of missing data, we incorporate missingness indicators into our models, and (4) to mitigate class imbalance, we use synthetic minority over-sampling techniques and to evaluate the contribution of each class to the overall performance, we use Index of Balanced Accuracy (IBA) score. We, therefore, believe that our study adds further evidence to the effectiveness and applicability of data collected during the first 24 hours of ICU admission in predicting discharge destination.

II. Methods

A. Dataset

The data for this study was obtained from the eICU Collaborative Research Database (eICU-CRD) [14], a multi-center critical care database supported by Philips Healthcare and the Laboratory for Computational Physiology [15] at the Massachusetts Institute of Technology. eICU-CRD comprises 200,859 ICU stays, from 166,355 hospital stays for 139,367 unique patients admitted to one of 335 ICUs at 208 hospitals across the United States between 2014 and 2015.

We included the 32 variables that are used to calculate the APACHE IV score. The predictor variables include patient demographics, chronic health condition, ICU admission diagnosis, and whether it is related to emergency surgery or not, admission source, and physiologic and laboratory variables from the first 24 hours of the ICU stay. A complete list of included variables is detailed in Table I. To manage the complexity of APACHE admission diagnosis data, we categorized the 407 unique admission diagnoses into 21 clinically meaningful categories provided by the eICU-CRD code repository [16].

B. Data Pre-Processing

1) Data Inclusion and Exclusion: All categorical data were coded into dummy variables using one-hot encoding and, to minimize the effect of previous ICU admissions for patients with multiple ICU stays, only the first stay was included in the analysis. However, the order of hospital admissions for a given patient cannot typically be discerned in eICU-CRD. For this reason, only the patients who have only one hospital admission and those who have multiple hospital admissions but the order of the admissions can be inferred were included in the cohort. For example, for patients with two ICU admissions, if the patient expires after one of these admissions, the other admission was included in the cohort. Also, if multiple ICU stays happened in different years, the first stay was included based on chronological order.

The discharge destination outcome in eICU-CRD is categorized into eight categories: death, home, nursing facilities (skilled nursing facility, nursing home), rehabilitation, other hospitals, other external, and other. Transfers to `other external` (n = 5,517, 3.3%), `other hospital` (n = 4,561, 2.7%), and `other` (n = 5,560, 3.3%) were excluded from the analysis since they were not a final disposition and since, due to the lack of a consensus definition for these destinations among the 208 hospitals included in eICU-CRD, they could correspond to different discharge destinations. These exclusions left 115,248 ICU stays for model development and further analysis.

2) Missing Data: From the 32 variables included in our study, 20 variables had missing values. Percent missing values ranged from 0.1% to 79%. To reduce the bias resulting from nonrandom missing data, we used the following two strategies, one at a time:

Imputation–Missing values were imputed using Multivariate Imputation by Chained Equations (MICE) [17], which preserves the relations in the data by developing the imputation model for each feature in the dataset separately.

Indication of Missingness–Motivated by experimental evidence on the usefulness of missingness indicators (MI) [18], [19] and, conversely, with the limitations of imputation techniques in improving the performance of predictive models based on electronic health records [20], we did not perform any imputation on missing continuous variables and, instead, added twenty

TABLE I

PATIENT CHARACTERISTICS FROM THE APACHE IV MODEL USED AS THE PRIMARY INDICATOR VARIABLES OF THE DEVELOPED PREDICTIVE MODELS IN THE PRESENT STUDY. DATA ARE MEAN (SD) OR NUMBERS (%). P-VALUES ARE GENERATED BY THE ONE-WAY ANOVA TEST. *P-VALUES ARE GENERATED BY THE χ^2 TEST. **P-VALUE IS GENERATED BY THE KRUSKAL WALLIS TEST

Variable	Missing Value (#)	Death	Home	Nursing Facility	Rehabilitation	P-value
N (%)		11792 (10%)	79946 (69%)	17828 (16%)	5682 (5%)	
Age, years, median [Q1,Q3]	4062	71 [6,79]	61 [49,71]	73 [63,81]	68 [56,77]	<0.001**
Gender (Female)	42	5473 (46.5)	35277 (44.1)	9343 (52.4)	2551 (44.9)	<0.001*
Clinical Variables						
Mechanical Ventilation	–	0.5 (0.5)	0.2 (0.4)	0.3 (0.4)	0.3 (0.4)	<0.001
Dialysis	–	501 (4.2)	2342 (2.9)	699 (3.9)	95 (1.7)	<0.001*
GCS	1802	9.8 (4.8)	13.8 (2.7)	12.7 (3.2)	12.7 (3.4)	<0.001
Sodium (mEq/L)	22408	138.5 (7.3)	137.8 (4.9)	138.4 (6.2)	137.8 (5.1)	<0.001
Urine Output (mL/24h)	51211	1940.9 (1592.8)	1597.3 (1454.3)	1682.1 (1596.7)	1955.9 (4936.1)	<0.001
WBC ($\times 1000/mm^3$)	27053	15.7 (12.8)	11.6 (6.9)	12.8 (8.3)	12.5 (7.2)	<0.001
Temperature (°C)	4766	36 (1.7)	36.5 (0.7)	36.4 (0.9)	36.5 (0.9)	<0.001
Respiratory Rate (/min)	698	29.9 (14.8)	24.4 (15.0)	26.8 (15.0)	25.7 (15.2)	<0.001
Heart Rate (/min)	233	111.3 (35.2)	97.7 (30.2)	101.7 (30.3)	100.3 (30.2)	<0.001
Mean Blood Pressure (mmHg)	311	81.9 (47.7)	87.9 (40.1)	86.7 (43.9)	92.9 (44.0)	<0.001
Creatinine (mg/dL)	22584	2.1 (1.7)	1.4 (1.7)	1.6 (1.6)	1.4 (1.4)	<0.001
Arterial pH	89176	7.3 (0.1)	7.4 (0.1)	7.4 (0.1)	7.4 (0.1)	<0.001
Hematocrit (%)	24522	31.4 (7.8)	33.4 (6.8)	31.5 (6.5)	32.7 (6.8)	<0.001
Albumin (g/L)	70813	2.5 (0.7)	3.0 (0.7)	2.7 (0.6)	3.0 (0.7)	<0.001
pO_2 (mmHg)	89176	132.1 (94.2)	132.0 (82.5)	127.7 (81.6)	132.4 (83.4)	0.022
pCO_2 (mmHg)	89176	41.7 (14.7)	42.8 (12.3)	43.2 (13.4)	41.8 (11.4)	<0.001
FiO_2 (%)	89176	71.5 (26.7)	55.7 (25.3)	57.7 (25.2)	58.9 (25.7)	<0.001
Urea (mEq/L)	23083	38.6 (26.4)	22.7 (19.0)	31.1 (23.7)	24.7 (19.2)	<0.001
Blood Sugar Level (mg/dL)	13222	180.9 (113.2)	161.9 (101.6)	160.5 (91.0)	158.0 (85.3)	<0.001
Bilirubin (mg/dL)	74850	2.1 (4.3)	1.0 (1.8)	1.0 (1.7)	1.1 (1.9)	<0.001
Chronic Health Condition	–					
AIDS		19 (0.2)	78 (0.1)	15 (0.1)	2 (0.0)	0.061*
Hepatic Failure		318 (2.7)	1045 (1.3)	236 (1.3)	38 (0.7)	<0.001*
Lymphoma		93 (0.8)	281 (0.4)	91 (0.5)	27 (0.5)	<0.001*
Metastatic Carcinoma		514 (4.4)	1380 (1.7)	317 (1.8)	88 (1.5)	<0.001*
Leukemia/Myeloma		186 (1.6)	492 (0.6)	122 (0.7)	38 (0.7)	<0.001*
Immunosuppression		555 (4.7)	1871 (2.3)	449 (2.5)	112 (2.0)	<0.001*
Cirrhosis		378 (3.2)	1189 (1.5)	295 (1.7)	46 (0.8)	<0.001*
Admission Information	–					
Readmission		1079 (9.1)	2907 (3.6)	1367 (7.7)	445 (7.8)	
Pre-ICU LOS (days)		2.3 (9.9)	0.7 (2.7)	1.5 (4.4)	1.4 (3.9)	<0.001
Admission Source	167					<0.001*
Direct Admission		73 (0.6)	223 (0.3)	67 (0.4)	24 (0.4)	
Emergency Room		195 (1.7)	4229 (5.3)	656 (3.7)	253 (4.5)	
Floor		782 (6.6)	13925 (17.4)	2519 (14.1)	1117 (19.7)	
Operating/Recovery Room		10 (0.1)	237 (0.3)	8 (0.0)	11 (0.2)	
Other Admission Source		6178 (52.5)	43544 (54.5)	9366 (52.6)	2808 (49.5)	
Other Hospital		905 (7.7)	6056 (7.6)	1076 (6.0)	398 (7.0)	
Other ICU		383 (3.3)	1424 (1.8)	352 (2.0)	148 (2.6)	
Step-down Unit		3238 (27.5)	10195 (12.8)	3763 (21.1)	918 (16.2)	
Admission Diagnosis	–					
Thrombolytics		212 (1.8)	1791 (2.2)	97 (0.5)	48 (0.8)	<0.001*
Diagnosis Group						<0.001*
ACS		298 (2.5)	6694 (8.4)	435 (2.4)	113 (2.0)	
ARF		151 (1.3)	1036 (1.3)	320 (1.8)	61 (1.1)	
Asthma-Emphys		228 (1.9)	2344 (2.9)	548 (3.1)	97 (1.7)	
CABG		90 (0.8)	3799 (4.8)	512 (2.9)	269 (4.7)	
CHF		564 (4.8)	2900 (3.6)	754 (4.2)	161 (2.8)	
CVA		1010 (8.6)	4675 (5.8)	1400 (7.9)	1107 (19.5)	
CVOther		137 (1.2)	2683 (3.4)	283 (1.6)	94 (1.7)	
CardiacArrest		1958 (16.6)	4787 (6.0)	1004 (5.6)	240 (4.2)	
ChestPain		15 (0.1)	598 (0.7)	44 (0.2)	14 (0.2)	
Coma		189 (1.6)	1050 (1.3)	388 (2.2)	81 (1.4)	
DKA		30 (0.3)	3558 (4.5)	148 (0.8)	31 (0.5)	
GIBleed		482 (4.1)	4447 (5.6)	1002 (5.6)	200 (3.5)	
GIObstruction		148 (1.3)	638 (0.8)	257 (1.4)	48 (0.8)	
Neuro		129 (1.1)	2965 (3.7)	528 (3.0)	268 (4.7)	
Overdose		31 (0.3)	2242 (2.8)	123 (0.7)	52 (0.9)	
PNA		677 (5.7)	2000 (2.5)	846 (4.7)	143 (2.5)	
RespMed		1075 (9.1)	4329 (5.4)	1082 (6.1)	266 (4.7)	
Sepsis		2682 (22.7)	7016 (8.8)	3636 (20.4)	514 (9.0)	
Trauma		427 (3.6)	3024 (3.8)	1015 (5.7)	681 (12.0)	
ValveDz		60 (0.5)	2140 (2.7)	316 (1.8)	116 (2.0)	
Other		1411 (12.0)	17021 (21.3)	3187 (17.9)	1126 (19.8)	

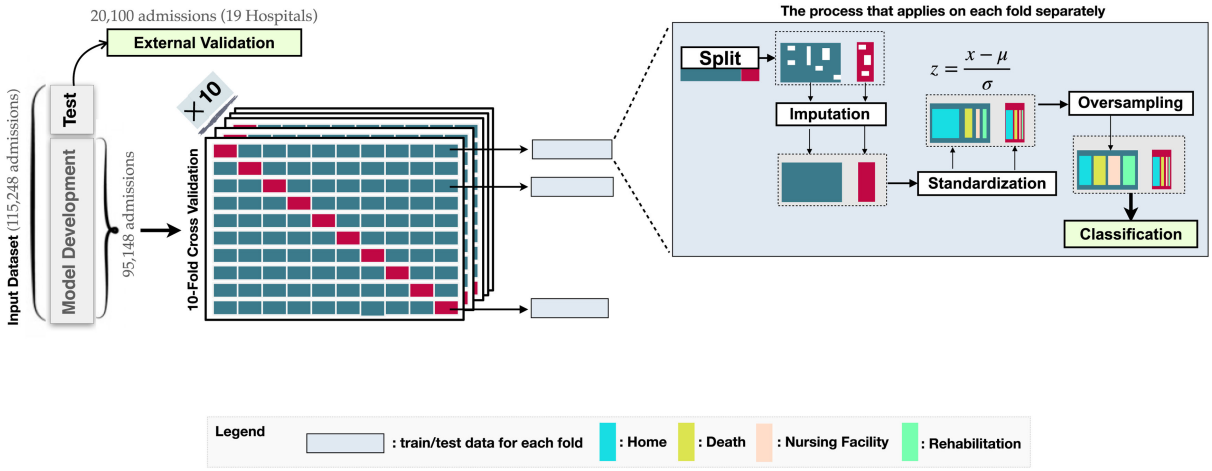


Fig. 1. The overall architecture of the applied machine learning cycle. During model development, we train each model 10 times, each time the *development* sample (n = 95,158) is partitioned into ten approximately equally sized sub-samples (n = 9,515).

auxiliary binary variables to indicate the missingness of these variables (e.g. 0: presence and 1: absence of measurements).

The continuous features of the resulting arrays were then standardized into z-scores by subtracting the mean and scaling each feature to unit variance.

3) Class Imbalance: Since the eICU dataset is highly imbalanced in terms of the distribution of discharge destinations (given in Table I), our predictive models tend to focus on the majority class. To mitigate this, we employed the following synthetic oversampling techniques when developing each stage of the cross-validation process (see Fig. 1).

SMOTE-NC–The Synthetic Minority Over-sampling TEchnique-Nominal Continuous (SMOTE-NC) [21] approach proposes oversampling the minority classes by creating synthetic samples based on feature-space (rather than data-space). Using this algorithm, the new samples are generated in the following way:

Given a minority set $\mathcal{S} = \{s_1, s_2, ..., s_N\}$, with each s_j representing a sample of the minority class, and a penalty factor M (the median standard deviation of all continuous features for the minority class), the SMOTE-NC approach finds the potential nearest-neighbor of each $s \in \mathcal{S}$, using:

$$d(s, s_j) = \sqrt{\sum_{i=1}^{n} (s_i - s_{ji})^2 + lM^2} \tag{1}$$

where $j \in \{1, ..., N\} \setminus \{i\}$, and n and l respectively denote the number of continuous features and the number of different nominal features between sample s and its potential nearest neighbors. After identifying the k-nearest neighbors, for each continuous feature s_c and its corresponding s_{ci}'s ($i \in \{1, 2, \ldots, k\}$), synthetic samples will be created as: $s_{new} = \langle \langle s_c + \lambda \times (s_{ci} - s_c) \rangle, \langle s_n \rangle \rangle$, where λ is a random number between 0 and 1 and helps increase the generalizability of the decision region of the minority class. Each nominal feature, included in s_n, will be populated by the most frequent category of the corresponding k-nearest neighbors.

SMOTEENN–Similar to SMOTE-NC, this approach implements a synthetic minority over-sampling by interpolating new samples between marginal outliers and inliers. Additionally, to create more well-defined class clusters, it applies the SMOTE-Tomek approach, which removes Tomek links [22]– all pairs of samples where they belong to different classes and are each other's nearest neighbors [23]. In contrast to SMOTE-NC, the SMOTEENN approach deploys only quantitative measures and does not differentiate well between continuous and categorical variables. We used this approach to study the role of categorical variables in the sampling process and to explore their impact on the discrimination ability of our classifiers.

In addition to the above oversampling techniques, we used the class-weight approach to incorporate the weight (W_y) of each class into the cost function, by assigning higher weights to minority classes and lower weights to majority classes. To calculate the proper weight for each class we used:

$$W_y = \frac{\overbrace{N}^{\#\text{of samples}}}{\#classes \times \underbrace{|y|}_{\#\text{samples in class } y}} \tag{2}$$

C. Development of Predictive Models

The overall workflow used for building predictive models in the present study is illustrated in Fig. 1. To ensure that our results are not biased towards a specific learning algorithm and to mitigate the risk of over-fitting, we developed and evaluated a representative set of standard machine learning classifiers, including generalized linear (Logistic Regression (LR)), kernel-based (Support Vector Machines (SVM)), decision-tree based (Random Forest (RF), AdaBoost, XGBoost, and ExtraTrees), and sample-based (K-Nearest Neighbours (KNN)) classifiers as well as a deep-learning-based classification model (a multilayer perceptron deep neural network (DNN), consisting of five 23-node fully-connected hidden layers with `relu` activation functions, terminating at an output layer with `softmax` activation and `adam` optimizer). The structure of the DNN

TABLE II
CHARACTERISTICS OF THE HOSPITALS INCLUDED IN THE VALIDATION SET

Characteristic	Teaching Status	
	False	True
n (%)	14 (74)	5 (26)
# of beds, n (%)		
100 - 249	5 (35.7)	
250 - 499	3 (21.4)	1 (20)
<100	4 (28.6)	
≥ 500	2 (14.3)	4 (80)
Region, n (%)		
Midwest	4 (28.6)	3 (60)
Northeast	1 (7.1)	
South	5 (35.7)	2 (40)
West	4 (28.6)	

discussed in this paper is based only on the case of the optimal results obtained with different configurations. Table III in the Supplementary Material section presents various network configurations and their corresponding results, using different networks sizes, dropout regularization, and batch normalization. Hyperparameters for each method were determined using a nested 10-fold cross-validation Bayesian Optimization [24] (listed in supplementary Table I). Compared to the random grid search [25] approach, Bayesian Optimization is significantly faster and finds better hyperparameters [24].

As illustrated in Fig. 1, to build and evaluate each of these models, we used 10-fold cross-validation with ten iterations. However, given that cross-validation may not always reflect the predictive ability of a model [26], we isolated 20,100 ICU admissions from 19 (10%) randomly selected hospitals for held-out validation. The characteristics of these hospitals are listed in Table II.

During the development of each fold, we first split the dataset into train and test subsets. This was followed by sequentially applying each of the imputation, standardization, and oversampling processes to the corresponding training set. At the end of each stage, each model's predictive performance was evaluated and compared using AUC, precision, recall, and F1 scores. Given the highly imbalanced nature of our test set (in terms of the discharge destination distribution) and to better interpret the performance results, we also calculated the Index of Balanced Accuracy (IBA) [27] score. IBA is a robust measure of the overall accuracy in imbalanced domains and can be calculated as:

$$IBA_{\alpha} = \overbrace{(1+\alpha \times (TP_{rate} - TN_{rate}))}^{\in[1-\alpha,1+\alpha]} \times TP_{rate} \times TN_{rate} \tag{3}$$

where $0 \leq \alpha \leq 1$ is used to weight the relationship between True Positives (TP) and True Negatives (TN). As both types of correct results (TP_{rate} and TN_{rate}) can effectively contribute to the discharge decision, and to minimize the negative impact of high TN_{rate} and low TP_{rate} values on the score of minority classes, we use $\alpha = 0.1$, which is in line with the experimental results presented in [27]. The IBA score can range from 0 to 1, with higher scores indicating better classifier performance. As the performance data obtained from LR, XGboost and RF classifiers were not normally distributed, we used non-parametric Kruskal-Wallis tests, followed by Wilcoxon's pairwise tests to evaluate the significance of the difference in the predictive performance of these models.

Moreover, to further investigate the performance of our predictive models in terms of the minority class, we incorporated these models into more tuned learning architectures: hierarchical classification, a classification ensemble, and a clustering-classification ensemble.

Hierarchical Classification–To study the impact of imbalanced testing set on the performance of our classifiers, we developed a top-down model, called Local Classification per Parent Node (LCPN) [28], using a predefined class hierarchy, as shown in Fig. 2(a). The LCPN approach defines a binary classifier for each node of the hierarchy and excludes siblings when training a specific node. For example, if the first level classifier assigns the discharge destination to `Death`, the second-level classifier will only train with `Home` and `¬Home` classes. This way, the majority classes will be excluded from both training and testing tasks, resulting in more balanced testing sets for the minority classes. To define the structure of hierarchical classifiers used in this study, we implemented all possible permutations of nodes (with `Death` and `¬Death` always in the first level) and selected the best performing configuration.

Classification Ensemble–To evaluate the impact of noisy features on our prediction results and to improve the robustness of the process of quantifying feature importance in our developed models, we used a stacked ensemble learning model presented in Fig. 2(a). The two base models selected for this architecture, XGBoost and ExtraTrees, are robust to noise for high dimensional data [29] and were used to identify the new set of features to be used as training data for the RF classifier.

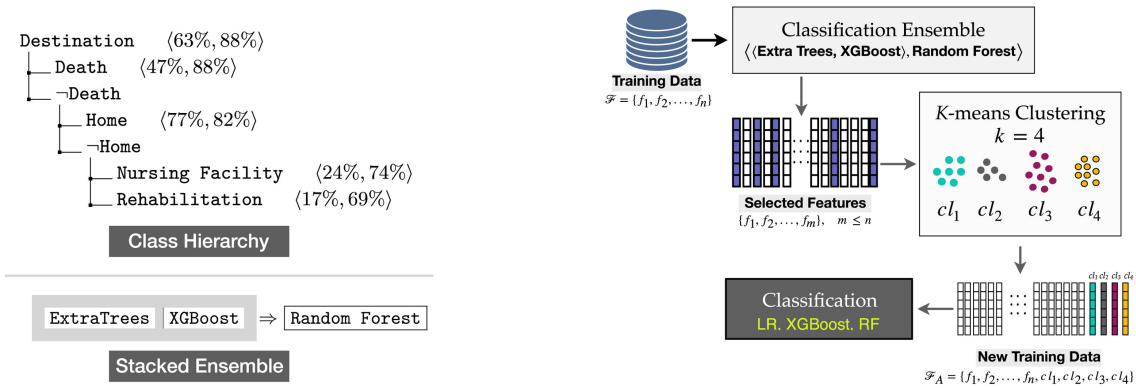
Clustering-Classification Ensemble (CCE)–The differences between training and testing distributions, due to oversampling, could cause concept drift during the training phase, which can be a reason for the relatively low performance of our models for the minority classes. To address this and to generate a more consolidated classifier, we developed an ensemble learning model that combines the above classification ensemble and the K-means clustering technique. In this framework, as illustrated in Fig. 2(b), we use the selected features by the previously induced classification ensemble as the input to the clustering layer (i.e., $\{f_1, f_2, ..., f_m\}, \quad m \leq n$). From this point of view, the cluster ensemble provides additional context-sensitive constraints for the main classification task, with the rationale that similar objects are more likely to share the same class label [30]. Any of the final classifiers (i.e. LR, RF, or XGBoost) takes as input the augmented feature set $\mathcal{F}_A = \{f_1, f_2, ..., f_n, cl_1, cl_2, cl_3, cl_4\}$, where cl_i is a binary variable and represents the relative co-occurrence of two data points in the same cluster [31].

All the computations and models were implemented using Python 3 with TensorFlow 2.0 [32], Keras [33], and Scikit-learn [34] libraries. To facilitate the replication of our study, the code repository of this study is publicly available on GitHub: https://github.com/data-intelligence-for-health-lab/Discharge-Prediction_eICU-CRD.

TABLE III
COMPARISON BETWEEN THE PERFORMANCE OF LR, RF, AND XBOOST ALGORITHMS IN PREDICTING DISCHARGE DESTINATION, USING EICU-CRD

	Precision*			Recall			Specificity			F1			AUC			IBA**		
Unbalanced	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF
Death	57%	61%	60%	42%	49%	46%	96%	96%	96%	48%	54%	52%	88%	90%	89%	38%	45%	42%
Home	76%	78%	76%	96%	95%	96%	30%	37%	32%	85%	85%	85%	82%	84%	82%	31%	37%	33%
Nursing Facility	40%	42%	41%	11%	18%	14%	97%	95%	96%	33%	24%	19%	74%	77%	75%	10%	16%	12%
Rehabilitation	0%	27%	3%	0%	1%	0%	100%	100%	100%	0%	1%	0%	70%	71%	70%	0%	1%	0%
Micro Avg	65%	69%	66%	73%	74%	73%	51%	55%	52%	67%	69%	67%	91%	92%	91%	27%	33%	29%
SMOTE-NC	Precision			Recall			Specificity			F1			AUC			IBA		
	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF
Death	42%	49%	45%	49%	54%	53%	92%	94%	93%	45%	51%	48%	85%	88%	86%	43%	49%	47%
Home	79%	80%	80%	77%	87%	83%	54%	50%	53%	78%	83%	82%	72%	80%	78%	42%	45%	45%
Nursing Facility	29%	36%	31%	18%	23%	27%	92%	93%	90%	22%	27%	28%	67%	73%	72%	15%	46%	22%
Rehabilitation	11%	15%	14%	24%	10%	8%	90%	97%	98%	15%	11%	10%	61%	66%	64%	21%	9%	7%
Micro Avg	65%	67%	67%	63%	70%	68%	65%	63%	65%	63%	68%	67%	84%	89%	88%	37%	39%	40%
SMOTE-NC, Missingness Indicator (MI)	Precision			Recall			Specificity			F1			AUC			IBA		
	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF
Death	45%	52%	48%	53%	59%	60%	93%	94%	93%	49%	55%	53%	86%	89%	89%	47%	54%	53%
Home	81%	82%	83%	73%	86%	83%	62%	57%	61%	77%	84%	83%	75%	82%	82%	46%	50%	52%
Nursing Facility	31%	37%	36%	25%	28%	34%	91%	92%	89%	27%	32%	34%	70%	75%	75%	21%	24%	28%
Rehabilitation	11%	18%	17%	29%	15%	11%	88%	96%	97%	15%	16%	12%	62%	67%	68%	24%	13%	9%
Micro Avg	67%	**70%**	70%	62%	**71%**	70%	71%	68%	70%	64%	**70%**	70%	84%	**90%**	89%	41%	45%	46%
SMOTEEN	Precision			Recall			Specificity			F1			AUC			IBA		
	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF
Death	35%	42%	39%	67%	67%	67%	86%	89%	88%	46%	52%	49%	87%	89%	88%	56%	58%	58%
Home	94%	91%	92%	37%	60%	52%	95%	86%	90%	53%	72%	67%	82%	83%	82%	33%	50%	45%
Nursing Facility	24%	27%	25%	38%	52%	57%	79%	76%	69%	29%	35%	34%	68%	73%	72%	29%	38%	39%
Rehabilitation	8%	13%	13%	50%	25%	22%	72%	92%	92%	14%	16%	15%	65%	68%	68%	35%	21%	19%
Micro Avg	74%	73%	73%	41%	57%	53%	90%	85%	87%	47%	62%	58%	62%	79%	73%	35%	48%	44%
Class Weight	Precision			Recall			Specificity			F1			AUC			IBA		
	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF	LR	XB	RF
Death	43%	61%	46%	57%	49%	61%	91%	96%	92%	49%	54%	53%	88%	90%	89%	51%	45%	54%
Home	83%	78%	84%	83%	95%	83%	61%	37%	63%	83%	85%	83%	82%	84%	82%	52%	37%	53%
Nursing Facility	35%	42%	35%	25%	18%	37%	92%	95%	87%	28%	24%	35%	74%	77%	75%	21%	16%	31%
Rehabilitation	16%	27%	24%	19%	1%	3%	92%	100%	99%	17%	1%	5%	70%	71%	69%	17%	1%	3%
Micro Avg	69%	69%	70%	69%	74%	70%	71%	55%	71%	68%	69%	69%	88%	92%	90%	46%	33%	47%

*Precision: Also known as Positive Predictive Value (PPV), **IBA: Index of Balanced Accuracy



(a) Hierarchical and ensemble classifiers. Numbers next to each node of the hierarchy present F1 and AUC metrics ⟨F1, AUC⟩

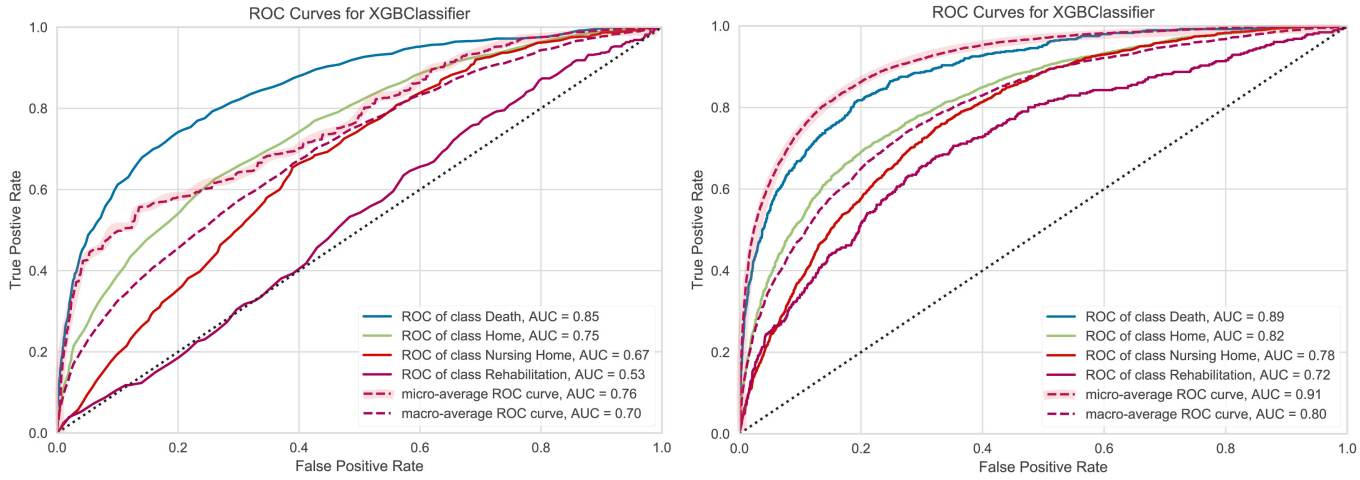
(b) Classification-Clustering Ensemble (CCE). As, in this study, we have four classes, the number of clusters is equal to four.

Fig. 2. An overview of the applied hierarchical, stacked ensemble, and CCE models used to predict discharge destination.

III. RESULTS

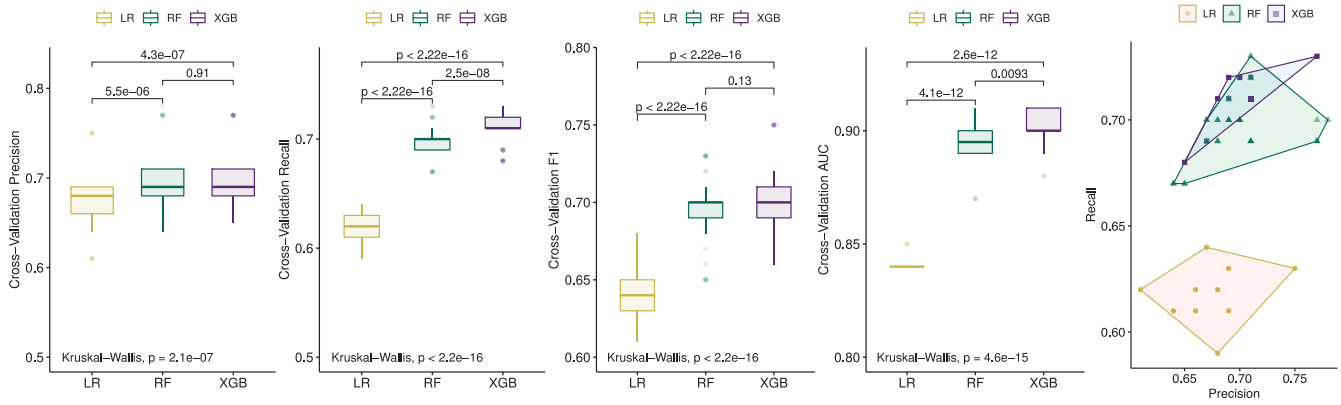
A. APACHE Score vs. APACHE Variables

To investigate the utility of the APACHE IV score in predicting hospital discharge destination in critically ill patients, and to compare it with that of the APACHE IV variables (listed in Table I), we tested each of the developed predictive models mentioned earlier on two different datasets: (1) a dataset consisting of only APACHE IV scores, and (2) the dataset listed in Table I, consisting of APACHE IV variables. Given that this score is a meta-feature that is calculated from a set of relevant basic features (listed in Table I), it holds information from these variables and could be informative enough for the multi-class classification task. Fig. 3(a) and (b) show the performance of the best performing model (XGBoost) in predicting discharge destination using both of these datasets. As illustrated in these figures, the APACHE IV score, compared to APACHE IV variables, performed quite poorly on the discrimination tasks, with relatively low AUC values for `Home`, `Nursing Home`, and `Rehabilitation` destinations. However, as APACHE IV score is designed for mortality prediction, the performance of the model trained on this score, in predicting the `Death` class



(a) Using only the APACHE IV score (b) Using all 32 APACHE IV variables

Fig. 3. (a) Comparison between the discrimination ability of APACHE IV score and (b) APACHE IV variables using XGB classifier. The macro-average of a metric weighs all classes equally, whereas the micro-average weighs classes based on their contribution (i.e. the number of instances) to the overall performance. Given the highly imbalanced nature of eICU-CRD, throughout this paper, we use the micro-average to report the average performance of the evaluation metrics.



(a) Precision (b) Recall (c) F1 (d) AUC (e) Convex-hull

Fig. 4. Comparison of classification performance among the top-three prediction models. Figure (e) combines figures (a) and (b) and intuitively compares the smallest convex set that encloses all pairs of (precision, recall) for the top-three classifiers.

(Fig. 3(a)), is comparable with that of the model trained on the APACHE IV variables (Fig. 3(b)). This could be because mortality and hospital discharge destination are two distinct clinical outcomes, requiring distinct predictor variables and predictive models.

B. Prediction Performance

From the seven predictive models developed and tested, KNN performed the worst across all measures for all four classes (average AUC: 55%, F1: 47%, precision: 66%, recall: 39%, IBA: 29%). The sequential deep neural network (DNN) produced the second worst results with an average F1 of 50%. From the remaining five models, the top-three classifiers were LR, RF, and XGBoost. Table II reports the average performances of these models across all 100 test sets (10 × 10-fold CV). In addition to the evaluation results listed in Table II, Fig. 4 confirms the superior performance of XGBoost over other models. It is readily seen that this model significantly outperformed LR and RF with higher recall, F1, and AUC scores. Further, to better understand and visualize the comparison between the performance of these classifiers in terms of their precision/recall, we plotted a two-dimensional convex hull shown in Fig. 4(e). The convex hull of a set of points intuitively visualizes the smallest convex set that encloses all the points [35]. Looking at this Figure, we see that XGBoost outperforms the other two methods with higher precision and recall values (recall, mean: 0.71 (±0.01); precision, mean: 0.70 (±0.03)).

C. Evaluation on the Held-out Test Set

We tested the best performing model, SMOTE-NC/MI XGBoost, on the held-out dataset, yielding a performance similar to that achieved during the model development, with an overall

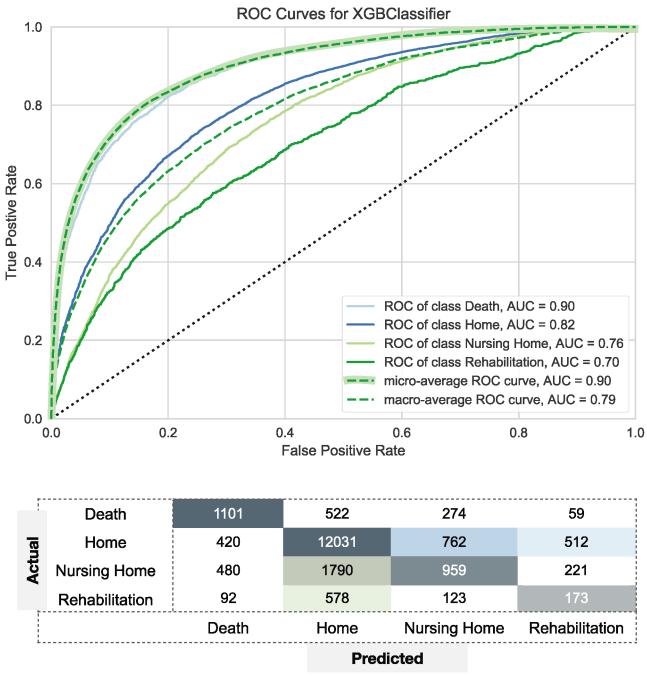


Fig. 5. Performance of XGBoost on the held-out dataset.

TABLE IV
PERFORMANCE OF THE BEST PERFORMING MODEL (XGBOOST) IN PREDICTING DISCHARGE DESTINATION FOR THE HELD-OUT DATASET

	Precision	Recall	Specificity	F1	AUC	IBA
Death	53%	56%	95%	54%	90%	51%
Home	81%	88%	55%	84%	82%	49%
Nursing Facility	45%	28%	93%	34%	76%	24%
Rehabilitation	18%	18%	96%	18%	70%	16%
Micro Avg	69%	71%	67%	69%	90%	44%

AUC of 90%, F1 of 69%, and recall of 71% (see Fig. 5 and Table IV). This confirms the discriminatory ability of the SMOTE-NC/MI XGBoost model for early prediction of hospital discharge destination for critically ill patients and confirms that the model can generalize to previously unseen ICU admission data.

IV. DISCUSSION

The main goal of this study was to develop machine learning models to accurately predict discharge destination for critically ill patients and to study the efficacy of the APACHE IV score and its corresponding variables as predictors of this outcome. From the results obtained in this study, we found that:

Finding1–The APACHE IV variables had a significantly better discrimination ability in predicting hospital discharge destination than the APACHE IV score alone. This reflects that mortality and discharge destination are two distinct clinical outcomes, requiring independent and customized prediction models.

Finding2–XGBoost yielded better discrimination performance than SVM, DNN, AdaBoost, ExtraTrees, LR, KNN, and RF. The ability of the XGBoost classifier to utilize redundant features and to model complex interactions and non-linear relations between features largely explains its superiority as a predictive model.

Finding3–The performance of prediction models was generally better when missingness indicators were incorporated into our models, indicating the informativeness of the presence/absence of predictor variables.

Finding4–While `Home` and `Death` discharges can be easily predicted using the APACHE IV variables collected during the first 24 hours of ICU admission, despite all the class imbalance mitigation techniques we utilized, accurate prediction of discharge to a nursing facility or rehabilitation is more challenging and may require more information than the APACHE IV variables (See the confusion matrix in Fig. 5). This could be because decisions about the nursing facility and rehabilitation discharges are influenced by a complex interplay of factors, including patient preferences, caregiver preferences, and organizational factors such as resource demand, staffing, bed capacity, and other operational factors [7].

While mortality remains an important outcome following critical illness, more nuanced outcomes reflecting the quality of life after hospital discharge are increasingly seen as important as well. These include measures of physical mobility, cognitive ability, mood, and capacity to return to work. These so-called patient-important outcomes also include discharge destination; in general, it is important for patients and their caregivers to know whether they will be able to return to independent living or will require increased supports in a care facility [4], [8]. Hospital discharge destination prediction, therefore, plays an essential role in decision-making and planning in the course of an ICU stay.

Discharge destination has implications for medical resource allocation, costs, and improving patients' potential for meaningful rehabilitation [36]. The ability to accurately predict discharge destination is, therefore, of potential value, especially if predictions can be made with sufficient lead time. Given the predictive information about the discharge destination, the ICU team could better plan for the allocation of limited bed sources and make arrangements with downstream care facilities. This is useful for clinicians, patients, families, and policymakers. For assessing the impact of different methods to tackle data imbalance, from the results presented in Table II, we observe that using synthetic oversampling techniques (i.e. SMOTE-NC, SMOTEENN) resulted in substantially higher `IBA` scores for the minority class (i.e. rehabilitation) than any other configuration. Given that the `class_weight` in logistic regression is applied to the `loss` function, compared to the `gini` and `entropy` functions in random forest and XGBoost, the results of the logistic regression classifier using `class_weight` is comparable to the over-sampled scenarios. Fig. 6(a) and (b) clearly illustrate the importance of using the IBA score as a metric to measure the behavior of a classifier with heavily skewed data distributions. For example, the AUC score for both `No-adjustment` and `Class_weight` configurations is higher than that of the over-sampled cases, while they perform poorly to predict the minority class with quite low precision, recall, F1, and AUC scores.

Concerning the oversampling techniques, while SMOTEENN resulted in high scores on IBA, classifiers that used this algorithm

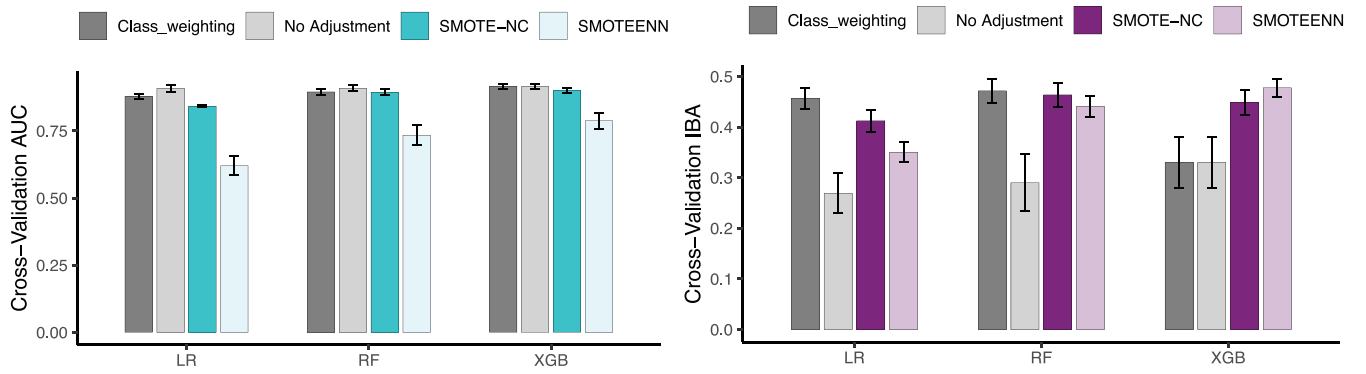


Fig. 6. Comparison of classification performance using different models and imputation techniques, as measured by the area under the receiver operating characteristic curve (AUC) and Index of Balanced Accuracy (IBA).

to balance their training set have a poor discrimination ability to predict discharge to `Home` (i.e., the majority class), with an average recall of 60% and F1 of 72% for XGB (with the highest IBA score). This increase in the number of False Negatives (FN) could be because of: (1) the failure of this method to differentiate between continuous and categorical data, and (2) the high frequency of Tomek links, due to the non-clusterable nature of the dataset in terms of discharge destination, and consequently the elimination of a large portion of samples from `Home` category. While a large number of FNs for Home discharges can be an acceptable outcome from clinical perspectives, these errors can lead to exposure to ineffective and unnecessary follow-up treatments that can have lasting side effects on the lives of patients and potentially cause costly delays in patients' recovery process. Thus, we exclude the prediction results of SMOTEENN-classifiers from further discussion and focus on the results of SMOTE-NC models.

With respect to missing data, looking at Table II, we see that including missingness indicators improved all the performance scores in all predictive models, with the highest impact on XGBoost's performance. Moreover, following the results of the Wilcoxon matched-pairs tests, SMOTE-NC/MI XGBoost statistically significantly outperforms SMOTE-NC XGBoost according to all six evaluation metrics. The missingness Indicator (MI) was paired with all the applied methods in this study. As XGBoost, with/without MI, outperforms the other approaches, we only report the results of SMOTE-NC/MI XGBoost (see supplementary Fig. 1). To further study the importance of missingness indicators as potential predictors, we calculated the mean feature importance across all classifiers listed in Table II by averaging the importance of each feature over all iterations of the 10-fold cross-validation. Interestingly, indicators for `elective surgery`, and `dialysis` were among the top-ten predictors, implying the marked impact of information about the presence or absence of predictor variables. This is in line with a recent study [19] in which the inclusion of missingness indicators significantly improved the results of mortality predictions using ICU data. Likewise, in a recent study, Tomašev *et al.* [18] used the patterns of missingness in a dataset to train their predictive models and achieved 90.2% accuracy for early prediction of acute kidney injury that required subsequent administration of dialysis.

In addition, to study the impact of imbalanced testing set on the performance of our models, we implemented the hierarchical classifier illustrated in Fig. 2(a). Using this model, from the top-three models reported in Table II, LR outperformed the other two models with higher F1 and AUC scores for all four classes. However, the SMOTE-NC/MI XGBoost still outperforms the best-performing hierarchical classifier, indicating that the underlying training procedure of this model is robust enough to our unbalanced test set. Moreover, following the results of 10-fold cross-validation for classification ensemble presented in Fig. 2(a) (average AUC: 85%, precision: 66%, recall: 71%, F1: 68%, and IBA: 35%), the SMOTE-NC/MI XGBoost and RF classifiers still outperform the ensemble model, which implies the developed XGBoost and RF can handle noisy features properly on their own. Finally, with respect to the CCE architecture presented in Fig. 2(b), while both SMOTE-NC/MI XGBoost and the best performance of the CCE model are comparable with an average AUC of 90% and 84% respectively, SMOTE-NC/MI XGBoost still outperforms CCE with higher true positives, true negatives as well as higher F1 and IBA scores. This could be due to a lack of clear clusters in the eICU dataset in terms of the discharge distribution. To further analyze this and to explore patterns of discharge destinations in eICU-CRD, across all continuous variables and categorical variables (i.e., gender and admission source), we visualized a random sample of the dataset, using Parallel [37] and Sankey [38] visualization techniques, respectively (see supplementary Figs. 2 and 3).

Several limitations should be noted. First, this was a retrospective study and our predictive models are based on association rather than causation. However, given the large sample size and the diversity of hospitals and patients included in eICU-CRD, the results of this study are likely robust. Second, to predict hospital discharge destination, we only used APACHE IV variables and did not study the utility of other variables such as prior visit numbers, ICU/hospital length of stay, treatment, and patients' past history; whether or not the integration of new variables into our predictive models can improve their performance still needs to be further investigated. Third, many patients whose

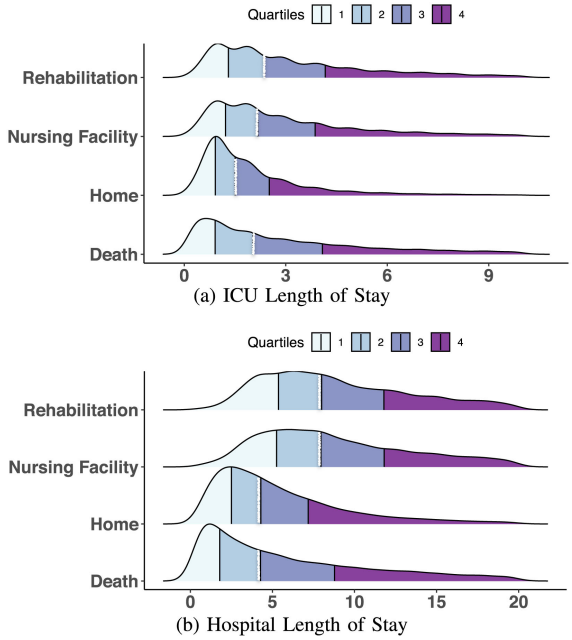


Fig. 7. The distribution of length of stay for different discharge destinations. The vertical white line shows the median of length of stay for each category.

ICU stay was less than 24 hours were not included in this study because the APACHE IV score is calculated based on the data collected within the first 24 hours after ICU admission. Also, 10% (15,638) of patients who were discharged to other external destinations (e.g. other hospitals) were not included in this study since they were not disposed to a final destination. Also, due to the lack of a consensus definition for these destinations in eICU-CRD, they could correspond to different discharge destinations. Finally, missing values in our dataset were of concern when training the classifiers. To address this, we added missingness indicators to investigate potential patterns of missing values. Nonetheless, we believe the prediction models developed in this study can improve the discharge planning process by mitigating the bias in decision making about discharge destination and by further exploring the intricate relationships between predictor variables and, therefore, better exploiting the richness of the data held in eICU-CRD.

Future work should study whether and how the information resulting from our predictive models helps to reduce patients' hospital length of stay while advancing the quality of care. As illustrated in Fig. 7, the average lengths of ICU and hospital stay for patients who are discharged to `Home` are two days and five days, respectively. These numbers for `Nursing Facility` and `Rehabilitation` discharges are three days and nine days, implying that developing accurate models to predict discharge destination with sufficient lead time could help the ICU team to better plan for dispositions and allocation of limited resources in critical care units. Also, we recommend future research to focus on the prospective evaluation of the predictive models developed in this study, as well as evaluation of the clinical impact of prediction modeling on clinical workflows and patient outcomes.

REFERENCES

[1] "What is critical care?" [Online]. Available: https://www.criticalcareontario.ca/EN/AboutUs/Pages/What-is-Critical-Care.aspx, Accessed on: Mar. 3, 2019.
[2] N. A. Halpern, D. A. Goldman, K. S. Tan, and S. M. Pastores, "Trends in critical care beds and use among population groups and medicare and medicaid beneficiaries in the united states: 2000–2010," *Crit. Care Medicine*, vol. 44, no. 8, pp. 1490–149, 2016.
[3] A. J. Campbell, J. A. Cook, G. Adey, and B. H. Cuthbertson, "Predicting death and readmission after intensive care discharge," *Brit. J. Anaesthesia*, vol. 100, no. 5, pp. 656–662, 2008.
[4] O. Badawi and M. J. Breslow, "Readmissions and death after ICU discharge: Development and validation of two predictive models," *PloS One*, vol. 7, no. 11, p. e48758-1–e48758-7, 2012.
[5] W. K. Barsoum *et al.*, "Predicting patient discharge disposition after total joint arthroplasty in the United States," *J. Arthroplasty*, vol. 25, no. 6, pp. 885–892, 2010.
[6] V. Agarwal, M. P. McRae, A. Bhardwaj, and R. W. Teasell, "A model to aid in the prediction of discharge location for stroke rehabilitation patients," *Arch. Physical Medicine Rehabil.*, vol. 84, no. 11, pp. 1703–1709, 2003.
[7] R. W. Mauthe, D. C. Haaf, P. Haya, and J. M. Krall, "Predicting discharge destination of stroke patients using a mathematical model based on six items from the functional independence measure," *Arch. Physical Medicine Rehabil.*, vol. 77, no. 1, pp. 10–13, 1996.
[8] J. E. Zimmerman, D. P. Wagner, E. A. Draper, and W. A. Knaus, "Improving intensive care unit discharge decisions: Supplementing physician judgment with predictions of next day risk for life support," *Crit. Care Medicine*, vol. 22, no. 9, pp. 1373–1384, 1994.
[9] M. W. Temple, C. U. Lehmann, and D. Fabbri, "Predicting discharge dates from the nicu using progress note data," *Pediatrics*, vol. 136, no. 2, pp. e395–e405, 2015.
[10] D. Cuadrado *et al.*, "Pursuing optimal prediction of discharge time in ICUS with machine learning methods," in *Proc. Conf. Artif. Intell. Medicine Europe*, 2019, pp. 150–154.
[11] C. R. Szubski *et al.*, "Predicting discharge to a long-term acute care hospital after admission to an intensive care unit," *Amer. J. Crit. Care*, vol. 23, no. 4, pp. e46–e53, 2014.
[12] K. Brook, C. A. Camargo, K. B. Christopher, and S. A. Quraishi, "Admission vitamin D status is associated with discharge destination in critically ill surgical patients," *Ann. Intensive Care*, vol. 5, no. 1, pp. 23-1–23-9, 2015.
[13] W. E. Muhlestein *et al.*, "Using a guided machine learning ensemble model to predict discharge disposition following meningioma resection," *J. Neurolog. Surgery Part B: Skull Base*, vol. 79, no. 02, pp. 123–130, 2018.
[14] T. J. Pollard, A. E. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi, "The eICU collaborative research database, a freely available multicenter database for critical care research," *Sci. Data*, vol. 5, pp. 180178-1–180178-5, 2018.
[15] A. L. Goldberger *et al.*, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.
[16] T. J. Pllard *et al.*, "Mit-lcp/eicu-code: eicu-crd code repository," [Online]. Available: https://doi.org/10.5281/zenodo.1249016, 2018.
[17] S. V. Buuren and K. Groothuis-Oudshoorn, "MICE: Multivariate imputation by chained equations in R," *J Statistical Softw.*, pp. 45–3, pp. 1–67, 2010.
[18] N. Tomašev *et al.*, "A clinically applicable approach to continuous prediction of future acute kidney injury," *Nature*, vol. 572, no. 7767, pp. 116–119, 2019.
[19] A. Sharafoddini, J. A. Dubin, D. M. Maslove, and J. Lee, "A new insight into missing data in intensive care unit patient profiles: Observational study," *J. Med. Internet Res.* , vol. 7, no. 1, Jan. 2019, Art. no. e11605.
[20] N. Razavian and D. Sontag, "Temporal convolutional neural networks for diagnosis from lab tests," 2015.
[21] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.
[22] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explorations Newslett.*, vol. 6, no. 1, pp. 20–29, 2004.
[23] I. Tomek, "Two Modifications of CNN," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, no. 2, pp. 679–772, Nov. 1976.

[24] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2951–2959.
[25] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, Feb. 2012.
[26] E. W. Steyerberg and F. E. Harrell, "Prediction models need appropriate internal, internal–external, and external validation," *J. Clin. Epidemiol.*, vol. 69, pp. 245–247, 2016.
[27] V. García, R. A. Mollineda, and J. S. Sánchez, "Index of balanced accuracy: A performance measure for skewed class distributions," in *Pattern Recognition and Image Analysis*, H. Araujo, A. M. Mendonça, A. J. Pinho, and M. I. Torres, Eds. Berlin, Germany: Springer-Verlag, 2009, pp. 441–448.
[28] C. N. Silla and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining Knowl. Discovery*, vol. 22, no. 1-2, pp. 31–72, 2011.
[29] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, Apr. 2006.
[30] A. Acharya, E. R. Hruschka, J. Ghosh, and S. Acharyya, "C 3 e: A framework for combining ensembles of classifiers and clusterers," in *International Workshop on Multiple Classifier Systems*. Berlin, Germany: Springer-Verlag, 2011, pp. 269–278.
[31] A. Strehl and J. Ghosh, "Cluster ensembles—A knowledge reuse framework for combining multiple partitions," *J. Mach. Learn. Res.*, vol. 3, no. Dec, pp. 583–617, 2002.
[32] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th {USENIX} Symp. Operating Syst. Des. Implementation*, 2016, pp. 265–283.
[33] F. Chollet *et al.*, "Keras: The python deep learning library," *Astrophysics Source Code Library*, 2018. [Online]. Available: https://keras.io/, Accessed on: Aug. 2019.
[34] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
[35] C. B. Barber, D. P. Dobkin, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Trans. Math. Softw.*, vol. 22, no. 4, pp. 469–483, 1996.
[36] T. Williams and G. Leslie, "Delayed discharges from an adult intensive care unit," *Australian Health Rev.*, vol. 28, no. 1, pp. 87–96, 2004.
[37] A. Inselberg, *Parallel Coordinates*. Berlin, Germany: Springer-Verlag, 2009.
[38] P. Riehmann, M. Hanfler, and B. Froehlich, "Interactive sankey diagrams," in *Proc. IEEE Symp. Inf. Visualization*, 2005, pp. 233–240.