Artificial Intelligence In Medicine 113 (2021) 102032

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Position Paper

Multimodal tensor-based method for integrative and continuous patient monitoring during postoperative cardiac care

Larry Hernandez [a,1], Renaid Kim [a,1], Neriman Tokcan [b], Harm Derksen [c], Ben E. Biesterveld [d], Alfred Croteau [e], Aaron M. Williams [d], Michael Mathis [f], Kayvan Najarian [a,g,h,i], Jonathan Gryak [a,i,*]

[a] Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI 48109, United States
[b] The Eli and Edythe L. Broad Institute of MIT and Harvard, Cambridge, MA 02142, United States
[c] Department of Mathematics, University of Michigan, Ann Arbor, MI 48109, United States
[d] Department of Surgery, University of Michigan, Ann Arbor, MI 48109, United States
[e] Hartford HealthCare Medical Group, Hartford, CT 06106, United States
[f] Department of Anesthesiology, University of Michigan, Ann Arbor, MI 48109, United States
[g] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109, United States
[h] Michigan Center for Integrative Research in Critical Care (MCIRCC), University of Michigan, Ann Arbor, MI 48109, United States
[i] Michigan Institute for Data Science (MIDAS), University of Michigan, Ann Arbor, MI 48109, United States

ARTICLE INFO

Keywords:
Hemodynamic decompensation
Postoperative care
Tensor methods
Multimodal analysis

ABSTRACT

Patients recovering from cardiovascular surgeries may develop life-threatening complications such as hemodynamic decompensation, making the monitoring of patients for such complications an essential component of postoperative care. However, this need has given rise to an inexorable increase in the number and modalities of data points collected, making it challenging to effectively analyze in real time. While many algorithms exist to assist in monitoring these patients, they often lack accuracy and specificity, leading to alarm fatigue among healthcare practitioners.

In this study we propose a multimodal approach that incorporates salient physiological signals and EHR data to predict the onset of hemodynamic decompensation. A retrospective dataset of patients recovering from cardiac surgery was created and used to train predictive models. Advanced signal processing techniques were employed to extract complex features from physiological waveforms, while a novel tensor-based dimensionality reduction method was used to reduce the size of the feature space. These methods were evaluated for predicting the onset of decompensation at varying time intervals, ranging from a half-hour to 12 h prior to a decompensation event. The best performing models achieved AUCs of 0.87 and 0.80 for the half-hour and 12-h intervals respectively. These analyses evince that a multimodal approach can be used to develop clinical decision support systems that predict adverse events several hours in advance.

1. Introduction

Patients recovering from cardiovascular surgeries are at risk of developing life-threatening complications [1]. These complications often manifest as some form of hemodynamic instability such as arrhythmia, respiratory failure, and hypotension. Thus, the monitoring of patients for signs of hemodynamic instability is an essential component of postoperative care. The ability of clinicians to accurately predict future decompensations and other adverse events in postoperative care can greatly reduce patient mortality and improve overall patient outcomes.

The need for patient monitoring in postoperative care has resulted in the deployment of monitoring devices in intensive care units (ICUs) that often generate for each patient over 10,000 data points per second, including high density physiological waveform signals. While new biosensors have increased the number and quality of available physiological signals [2], the clinical utility of alarm algorithms essential to the timely detection or prediction of adverse conditions has continued to

* Corresponding author.
E-mail address: gryakj@med.umich.edu (J. Gryak).
[1] These authors contributed equally to the manuscript.

https://doi.org/10.1016/j.artmed.2021.102032
Received 5 February 2020; Received in revised form 6 January 2021; Accepted 8 February 2021
Available online 11 February 2021

advance at a slower pace than other medical technologies [3–8]. Moreover, multiple studies have shown that humans are poor at identifying physiological changes affecting more than two concurrent signals [9]. This means that even when ICU clinical teams are continuously monitoring a patient's physiological signals it may be challenging to recognize complex patterns that arise through some combination of changes across multiple signals [10].

Given the need to synthesize and make predictions on such data, numerous methods have been developed, which can be roughly dichotomized by their use of physiological signals or lack thereof.

A spectrum of existing predictive systems create recommendations based only on clinical data in the electronic health record (EHR) and/or occasional (non-continuous) measurement of physiological data. In [11], a predictive model was designed to track the changes in relative blood volume and heart rate during hemodialysis by using the ultrafiltration rate. The shock index, the ration of heart rate to systolic blood pressure, is widely used in emergency departments to predict mortality in traumatically injured patients [12]. In [13], activation functions for cardiovascular dynamic models were formulated, using electrocardiograms (ECGs) as the sole input. In [14], a fuzzy logic approach was developed to identify peripheral nerve disorders using EHR data. In [15], a machine learning approach for monitoring hemorrhage was proposed that assumed all test subjects had similar hemodynamic physiology. In [16], a machine learning approach used lab values, vital signs, and other EHR values for predicting hemodynamic instability in pediatric ICUs.

Other extant approaches that do incorporate information from physiological signals apply relatively simple measures defined using Heart Rate Variability (HRV) extracted from such signals to provide predictions on the hemodynamic state of patients [17–23]. These methods are primarily frequency-based metrics (i.e., high frequency (HF), low frequency (LF), and HF/LF ratio) [17,19,20]. For example, in [17], a predictive model was developed to identify high-risk hypertensive patients; however, this model's sensitivity was too low for use in a clinical study. Several studies throughout the last decade have shown the inadequacy of these metrics [18,24]. The insufficient accuracy of the aforementioned models may in part be attributed to their inability to analyze and integrate all streaming and static data produced by disparate monitoring systems along the continuum of a patient's treatment, i.e., from hospital admission and initial surgical insult through hospital discharge.

To address these concerns, in this study we propose a multimodal approach that incorporates salient physiological signals and EHR data to predict the onset of hemodynamic decompensation. In Section 2, we describe the construction of a retrospective dataset of patients recovering from cardiac surgery that was used to construct the prediction models. In Section 3, we describe the methods used for feature extraction, feature reduction, and model construction. Advanced signal processing techniques such as Taut String estimation and Dual-tree Complex Wavelet Packet Transform are employed to extract complex and non-obvious patterns from ECG signals, while a novel tensor-based dimensionality reduction method is used to reduce the size of the resultant feature space. In Section 3, we describe the results of these methods for predicting the onset of decompensation at varying time intervals, ranging from a half-hour to 12 h. The best performing models achieve an Area Under the Receiver Operating Characteristic curve (AUC) of 0.87 for the half-hour interval, with performance generally decreasing as the interval is increased to 12 h, with the best performing model at that interval achieving an AUC of 0.80. A discussion of the results and a conclusion follow in Sections 5 and 6.

2. Data

The data used in this study was collected retrospectively from Michigan Medicine data systems. The cohort includes patients who had elective cardiac surgery with cardiopulmonary bypass and whose physiological waveforms were captured to file. Cases that were emergent were excluded, as well as those that occurred on weekends or holidays as these tend to be atypical. A hand review of the surgical procedure text for each of these cases was performed to exclude cases involving chest washouts, ECMO cannulations, and emergency readmissions. This data collection yielded a total of 1863 prospective patients to include in our analysis.

Nine adverse events associated with hemodynamic instability were chosen by our clinical team for inclusion:

1. *Low Cardiac Index* – A new decrease in cardiac index below 2.0 L/min/m^2 with either no prior cardiac index measurement within 48 h; or prior cardiac index measurement within 48 h $\geq 2.0\,L/\mathrm{min}/m^2$.
2. *Sustained Low Mean Arterial Pressure* – Decrease in mean arterial pressure (MAP) below 55 mmHg for $\geq$ 120 min with either no prior MAP measurement within 48 h; or prior MAP measurement within 48 h $\geq$ 55. Note that if multiple sources of mean arterial pressure were being monitored, the highest mean arterial pressure measurement available was used.
3. *Mortality* – Patient death that occurred within six months after the end of their surgery.
4. *Epinephrine Bolus* – An injection of epinephrine may be indicated for a variety of cardiovascular conditions, including cardiac arrest and hypovolemia. A review of sample cases found that boluses of 10 mcg were significant, thus any epinephrine injections greater than this amount were considered.
5. *Inotropic Therapy Initiated* – Cardiac inotropes may be administered by clinical teams to improve the contractile strength of an acutely failing heart in order to maintain adequate blood flow to vital organs. After cardiac surgery, many patients receive some combination of inotropes intravenously. Additionally, new inotropic therapies may be initiated by clinical teams postoperatively in response to new hemodynamic decompensation. For the purposes of this study, the following medications were considered inotropes: milrinone, dobutamine, epinephrine, dopamine, and isoproterenol.
6. *Inotropic Therapy Escalated* $\geq$ 100% – In addition to tracking the introduction of new inotropic therapies, escalations of ongoing inotropic therapies were also considered. As the dosage of medications is frequently titrated up or down in small increments, a doubling of a medication was chosen as a proxy for a significant change in a patient's condition. However, the doubling of a medication's dosage must be above a pre-specified initial threshold to qualify as an event. Initial thresholds for each inotrope were: milrinone – 0.250 mcg/kg/min; dobutamine – 2.0 mcg/kg/min; epinephrine – 0.02 mcg/kg/min; dopamine – 2.5 mcg/kg/min; isoproterenol – 2.0 mcg/min.
7. *Vasopressor Therapy Initiated* – Vasopressors may be initiated by clinical teams to maintain adequate perfusion pressure of blood flow to vital organs. After cardiac surgery, many patients receive some combination of vasopressors intravenously. Additionally, new vasopressor therapies may be initiated by clinical teams postoperatively in response to new hemodynamic decompensation. For the purposes of this study, the medications vasopressin and norepinephrine were considered as vasopressors.
8. *Vasopressor Therapy Escalated* $\geq$ 100% – In addition to tracking the introduction of new vasopressor therapies, escalations of ongoing vasopressor therapies were also considered. As in the case for inotropic therapies, a doubling of a medication was used as a proxy for a significant change in a patient's condition. As before, the doubling of a medication's dosage must be above a pre-specified initial threshold to qualify as an event. Initial thresholds for each vasopressor were: vasopressin – 2 units/h; norepinephrine – 0.10 mcg/kg/min.
9. *Reintubation* – A patient was considered reintubated if, after having been removed from invasive mechanical ventilation support, support

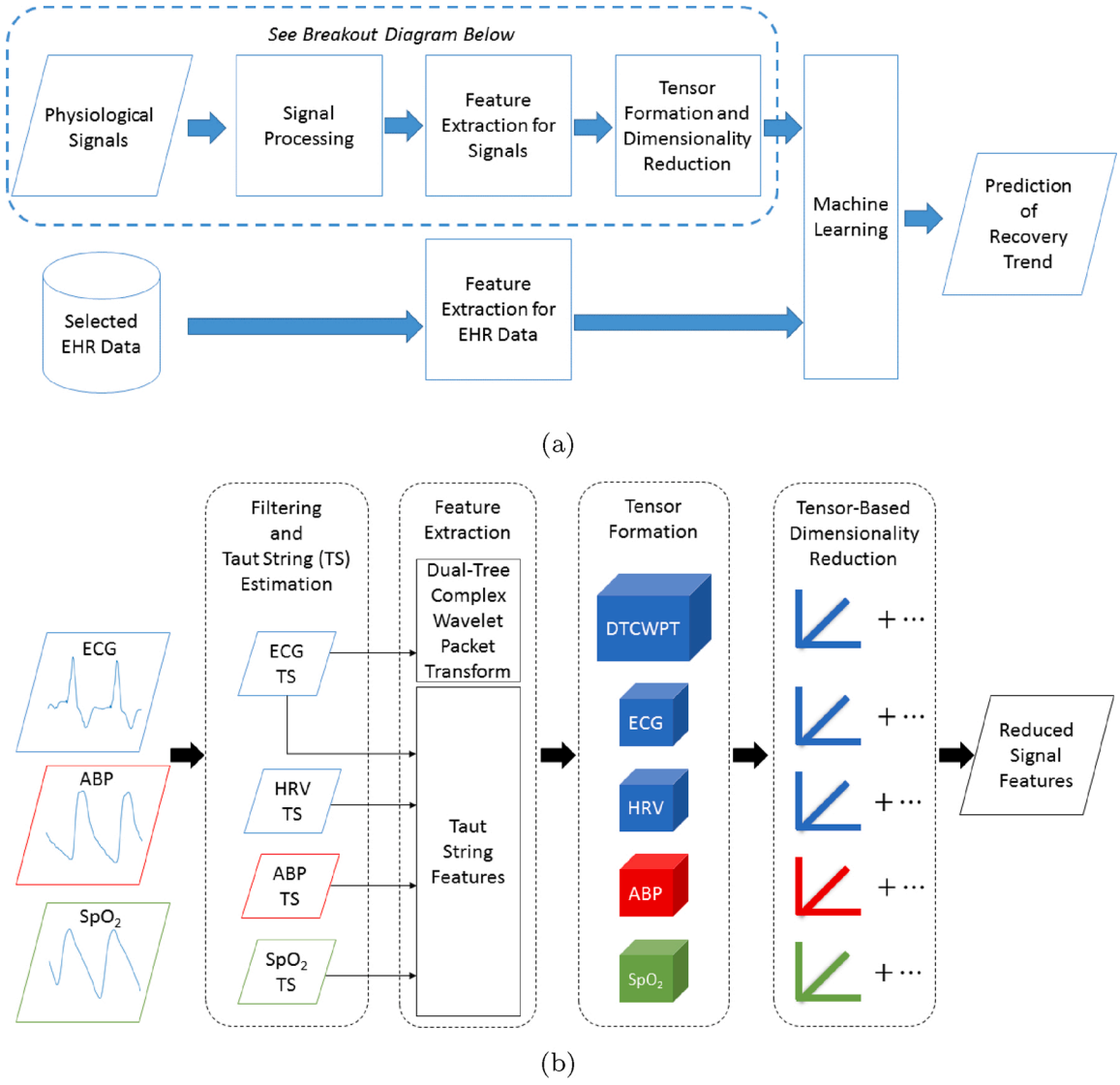

Fig. 1. (a) A depiction of the overall system, in which features extracted from physiological signals and EHR data are used to construct predictive models. (b) The proposed feature extraction and selection process for physiological signals.

was reinitiated via an invasive advanced airway (e.g., endotracheal tube).

As the patients included in this cohort were recovering from surgery, there is an initial period in which most patients are in a state of hemodynamic lability, during which vital signs and other measures of cardiovascular activity are in flux and clinical teams are actively mitigating known hemodynamic derangements. Therefore, the patient records of the prospective cohort were searched for instances of the above nine adverse events that occurred 24 h or later after the end of surgery. This search yielded 226 patients in the prospective cohort with one or more of the nine adverse events. In order to ensure that the adverse events discovered were genuine and due to hemodynamic decompensation, two clinicians verified each of the identified events, with a third clinician adjudicating any discrepancies that arose. After verification, there were 180 patients with at least one or more adverse events. This subpopulation comprises the *positive* class in the dataset. 226 patients from the prospective cohort that did not have any adverse events, as verified by clinical review, comprise the *negative* class in the dataset. The final dataset used in this study includes a total of 406 patients, averaging 67 years of age (min-max = 40–91 years), with 239 (59%) male and 167 (41%) female.

In constructing the training and validation datasets, patient data were aggregated so that approximately 35% of all samples were positive, while the remaining 65% were negative. The date and time of each adverse event were identified and analyzed for each sample in the positive class. For the negative class, a random time during each patient's observation period was chosen as the fiducial time point. A uniform distribution was used to determine these random times; this approach allowed the full spectrum of the recovery period to be captured in the analysis. Multiple fiducial time points were generated for a random selection of the negatives in order to ensure that this subpopulation comprised 65% of the total number of samples.

3. Methods

The proposed method for predicting hemodynamic decompensation is depicted in Fig. 1. Details about each step are provided in the subsections that follow.

3.1. Signal processing

The physiological signal waveforms utilized in this work were electrocardiogram (ECG), arterial blood pressure (ABP), and the

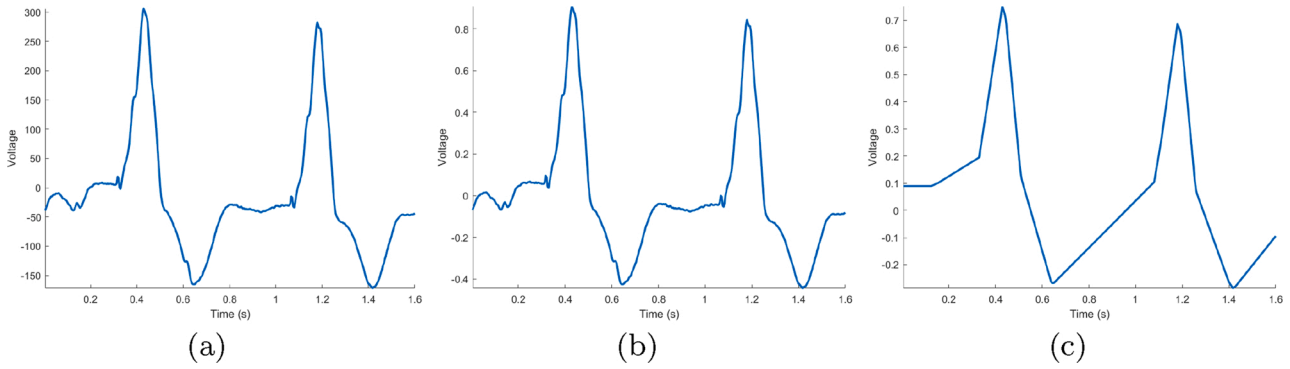

Fig. 2. An example of the filtering process used for ECG signals: (a) a sample of raw ECG signal; (b) the ECG sample after Butterworth filtering; and (c) a Taut String estimation of the sample ($\epsilon = 0.1575$).

photoplethysmogram obtained from the pulse oximeter used to calculate peripheral oxygen saturation (SpO_2). Relevant portions of each signal were extracted, after which multiple estimations of these signals were made using the Taut String method (Section 3.1.2). Further processing and feature extraction methods for each signal are described below.

3.1.1. Signal extraction: signal availability, prediction windows, tumbling analysis windows

For each fiducial time point, six 15-min *analysis windows* preceding the fiducial time point were identified for signal extraction and analysis. Associated with each 15-min "analysis window" is a "prediction window" preceding the fiducial time point. Prediction windows in this study were 0.5, 1, 2, 4, 8, and 12 h. Patients whose recorded signals were of insufficient duration for this analysis (<24 h, 45 min), or for whom discrepancies arose in signal duration greater than 15 s between the waveform timestamps from the EHR and duration as calculated based on the sampling rate, were discarded. This extraction and filtering process yielded $M = 243{,}246{,}223{,}191{,}160$, and 151 observations for prediction windows of 0.5, 1, 2, 4, 8, and 12 h, respectively, from 44 distinct positive and 97 negative patients (see Tables 5 and 6 in Appendix A for a breakdown of events by patient). Each 15-min analysis window was divided into five non-overlapping tumbling windows of 3-min duration. Tumbling windows of 3, 4, and 5 min in length were also tested, with 3-min windows yielding the best performance. In addition to better performance, a smaller window allows for a trained model deployed in a clinical setting to produce a prediction sooner. This may increase the impact of these predictions in alerting clinicians to patients who are transitioning from a stable to deteriorating state. Signal pre-processing and feature extraction methods were then applied to the physiological signal waveforms contained in each tumbling window, as described below.

3.1.2. Taut string

In [25], peak-based and statistical features calculated from Taut String (TS) estimation [26,27] of ECG signals were shown to be useful for detecting hemodynamic instability. Given a discrete signal $f = (f_1, \ldots, f_n)$, define the first-order finite difference as

$$D(f) = (f_2 - f_1, \ldots, f_n - f_{n-1}). \tag{1}$$

For a fixed $\epsilon > 0$, the Taut String estimate of f is the unique function g such that $\|f - g\|_\infty \leq \epsilon$ and $\|D(g)\|_2$ is minimal, with $\|\cdot\|_\infty$ and $\|\cdot\|_2$ denoting the max norm and Euclidean norm respectively. In general, g is a piecewise linear function whose segments can be intuitively visualized as a string pulled from both ends through $f + \epsilon$ and $f - \epsilon$.

In this study, the same features obtained from Taut String estimation in [25] were utilized for analysis of the windowed ECG signals.

3.1.3. Dual-tree complex wavelet packet transform

The authors in [25] also utilized statistical features calculated from the sets of coefficients obtained from application of the dual-tree complex wavelet transform (DTCWT) to each tumbling window. While DTCWT is useful in many applications, the decompositions it produces are non-optimal as the transform fails to consider the characteristics of the signal of interest. Therefore, in this study, an extended version of the DTCWT called the Dual-Tree Complex Wavelet *Packet* Transform (DTCWPT) [28] was utilized. DTCWPT provides a computationally efficient means for determining an optimal wavelet basis to faithfully represent the frequencies present in a given signal, while still preserving the approximate shift-invariance of DTCWT. This allows the transform to better capture complex features present in biomedical signals such as ECG.

Executing in parallel, the DTCWPT uses two wavelet filter banks to decompose, at each stage/level k, each of the subbands using both low-pass and high-pass perfect reconstruction (PR) filters banks (FBs). For the transform to achieve approximately analytic subbands, a necessary condition for approximate shift-invariance, the PR FBs must be selected so that the frequency response of each branch of the second wavelet packet FB is the discrete Hilbert transform of the corresponding branch of the first wavelet packet FB.

Let Ψ be the wavelet for the respective low-pass and high-pass filters $h_0(n)$ and $h_1(n)$, and Ψ' its Hilbert pair (i.e., $\Psi' = \mathcal{H}\{\Psi\}$). Given an orthonormal wavelet basis, the z-transforms of h_0 and h_1, denoted as H_0 and H_1, are related by the equation

$$H_1(e^{\mathrm{j}w}) = -e^{\mathrm{j}dw} H_0^*(e^{j(w-d)}), \tag{2}$$

while H_1 and H_1' have the relationship

$$H_1' = -j \cdot \mathrm{sgn}(w) e^{j0.5w} H_1(e^{\mathrm{j}w}), \tag{3}$$

where sgn is the signum function, d is an odd integer, and $|w| < \pi$.

Let $H^{(k)}(e^{\mathrm{j}w})$ be the equivalent response at stage/level k, then it can be shown [28] that:

$$H^{(k)}(e^{\mathrm{j}w}) = H_1(e^{j2(k-1)w}) \prod_{m=0}^{k-2} H_0(e^{j2\mathrm{m}w}) \tag{4}$$

The equivalent response of the second filter bank's corresponding branch is:

$$H'(k)(e^{\mathrm{j}w}) = -e^{j0.5w} \mathcal{H}\{H^{(k)}(e^{\mathrm{j}w})\}. \tag{5}$$

Corresponding filters within each FB must be chosen to be identical but can still be chosen to reflect the frequencies and morphology of the underlying signal to be analyzed. Increasing the number of stages/levels k yields greater frequency resolution provided the input signal is of sufficient length ($a \cdot 2^k, a \in \mathbb{N}$), albeit at greater computational expense.

Table 1
The features extracted from each signal.

Taut string estimates of ECG and HRV Number of line segments per heartbeat, Number of inflection segments per heartbeat, Total variation of noise per heartbeat, Total variation of denoised signal per heartbeat, Power of denoised signal, Power of noise (6 features per signal, 12 total)
DTCWPT of Taut String Estimate of ECG Standard deviation, Shannon entropy, Log Energy Entropy, Energy, Power, Min, Mean, Max, Median of largest 16, Range, Mean gradient, Kurtosis, Skewness, Complexity, Mobility, Log of variance of probability distribution, Mean value of amplitude of FFT, sum of auto-correlation sequence, mean value of cross-covariance (19 features)
Taut String Estimates of Arterial Blood Pressure and SpO_2 Total number of peaks; Min, Max, Mean, Median, Standard deviation of time interval between consecutive systolic peaks; Min, Max, Mean, Median, Standard deviation of time interval between systolic peak and subsequent diastolic peak; Min, Max, Mean, Median, Standard deviation of relative amplitude between consecutive systolic peaks; Min, Max, Mean, Median, Standard deviation of relative amplitude between systolic peak and subsequent diastolic peak (21 features per signal, 42 total)

Table 2
Numerical ranges and critical threshold values used to encode patient laboratory results.

Lab parameter	Reference range	Critical threshold	Unit
Creatinine	Male: 0.7–1.3 Female: 0.5–1.0	>2.0	mg/dL
Glucose	70–180	<40	mg/dL
Hematocrit (HCT)	Male: 40–50 Female: 36–48	<21	%
Hemoglobin (Hgb)	Male: 13.5–17 Female: 12–16	<7	g/dL
International normalized ratio (INR)	0.9 –1.2	>2.0	
Lactate	Arterial: 0.5–1.6 Venous: 0.5–2.2	>4.0	mmol/L
Platelet count (PLT)	150–400	<50	10^9/L
Potassium	3.5–5.0	>6.0	mmol/L
Sodium	136–146	>155	mmol/L
White blood cell count (WBC)	4–10	>20	10^9/L

In this study, $k = 2$ was used. Further details can be found in [28].

3.2. Feature extraction I: ECG-based predictors

The ECG signal within each tumbling window was band-pass filtered with a 2nd order Butterworth filter to remove artifacts that commonly arise in electrocardiograms. The cutoff frequencies employed with these clinical ECG signals were 0.5 and 40 Hz.

Taut String estimation was applied to the filtered ECG using five values of the parameter ϵ: 0.0100, 0.1575, 0.3050, 0.4525, and 0.6000. For each tumbling window and value of ϵ, six morphological and statistical predictive features were calculated from the resulting Taut String estimates. These features include: (1) number of line segments per heartbeat, (2) number of inflection segments per heartbeat, (3) total variation of the noise per heartbeat, (4) total variation of de-noised signal per heartbeat, (5) power of the de-noised signal, (6) and power of the noise. Fig. 2 illustrates the filtering and Taut String estimation process for a sample of ECG signal.

A second set of features was generated by applying the DTCWPT with two levels of decomposition to the Taut String estimates of the filtered ECG signal levels. 19 features were calculated from each set of coefficients, including Shannon entropy, the log energy entropy, energy, power, mean coefficient value, maximum coefficient value, minimum coefficient value, standard deviation, range, kurtosis, and skewness. With two levels of decomposition, each of the two filter banks yielded 4 sets of coefficients, thereby producing 152 wavelet features. These 152 features were calculated for each tumbling window and value of ϵ.

To generate the Heart Rate Variability (HRV) from the filtered ECG waveform, R peaks were first identified using a novel peak detection method as described in Appendix B. If $t = (t_1, \ldots, t_n)$ is the discrete signal representing the times at which the identified R peaks occurred, then the HRV signal is defined as $D(t)$ (see Eq. (1)). Five Taut String estimates of the HRV signal were obtained. The values of ϵ were 0.001, 0.0258, 0.0505, 0.0753, and 0.100. The six aforementioned morphological and statistical predictive features obtained from Taut String Estimation were then calculated for each value of ϵ.

3.3. Feature extraction II: arterial blood pressure

The arterial blood pressure (ABP) waveform of each tumbling window was band-pass filtered with a 3rd order Butterworth filter to remove artifacts. The cutoff frequencies were 1.25 and 25 Hz.

Taut String estimation was applied to the filtered waveform. The values of ϵ were 0.1, 0.7, 1.3, 1.9, and 2.5. The systolic peak and the diastolic peak within each cycle of the signal were identified [29]. From these two peaks, 21 features that have been shown to be effective for detecting changes in hemodynamic state [30] were calculated for each value of ϵ. These include the total number of peaks, as well as the min, max, mean, median, and standard deviation of (a) the time interval between consecutive systolic peaks, (b) the time interval between systolic peak and subsequent diastolic peak (c) the relative amplitude between consecutive systolic peaks, and (d) the relative amplitude between systolic peak and subsequent diastolic peak. These 21 features were calculated for each tumbling window and value of ϵ.

3.4. Feature extraction III: SpO_2 signal

The SpO_2 signal within each tumbling window was band-pass filtered with a 3rd order Butterworth filter to remove artifacts. The cutoff frequencies were 1.75 and 10 Hz. The feature extraction process for SpO_2 signals was identical to the process for the ABP signals. Just as with ABP, the process produced 21 features for each tumbling window and value of ϵ.

The features extracted from each signal are summarized in Table 1.

3.5. Feature extraction IV: electronic health records

In addition to signal-based features, electronic health record (EHR) data was incorporated into the predictive model. These data include the static EHR data, such as patient age, race, and 30 comorbidities such as obesity, diabetes, alcohol abuse, chronic pulmonary disease, and drug abuse; and the temporal EHR data, including the lab results (i.e., blood tests, hematology, metabolites, kidney function, immune function, coagulation, among others), medications administered during the course of a patient's stay (cardiovascular infusions, other cardiovascular drugs, non-cardiovascular drugs), daily respiratory support (fraction of inspired oxygen, positive end-expiratory pressure, intubation status), and vital signs not derivable from the physiological signals used in the study (e.g., temperature and oxygen saturation). Values for temporal EHR data were extracted for each tumbling window.

Clinical lab results were encoded into "critical", "low", "normal", "high" and "unknown" categories. Values categorized as critical may lie outside either the low or high value ranges, depending on the specific

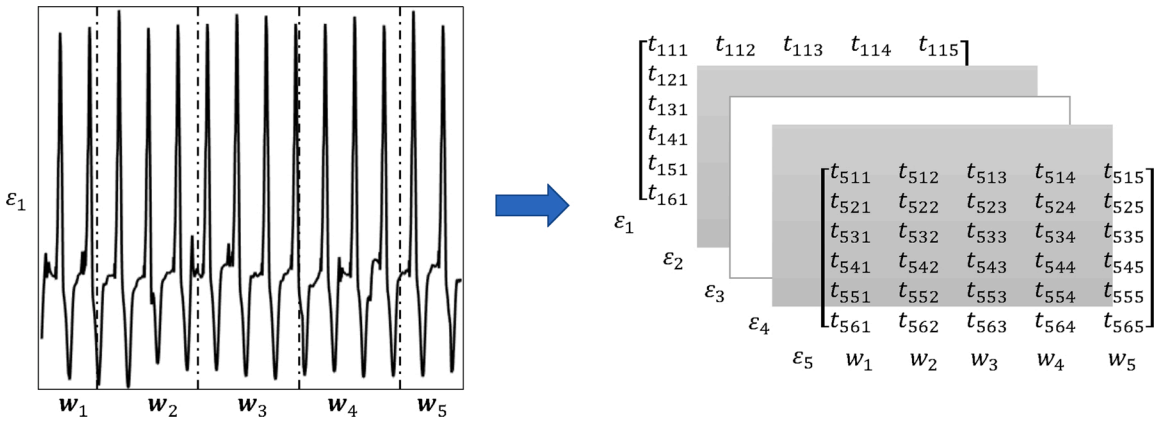

Fig. 3. Constructing a tensor from signal features. In this example, feature vectors are extracted from each tumbling window w_i (with $1 \leq i \leq 5$) from the Taut String estimate of the ECG signal using parameterϵ_1, producing a 6×5 feature matrix. This process is repeated for each value of ϵ, with the resultant matrices stacked together to form a third-order tensor of dimension $5 \times 6 \times 5$.

laboratory parameter. The thresholds for critical values were chosen to target a cardiovascular ICU population. Reference ranges and critical thresholds for these lab values are presented in Table 2. In total, 101 features were selected from electronic health records.

3.6. Tensor-based dimensionality reduction

The feature extraction process yields 5150 features from the ECG, ABP, and SpO_2 waveforms when all tumbling windows and values of ϵ are taken into consideration. Typically in machine learning applications these features would be concatenated into a one-dimensional feature vector, after which various methods of feature selection, such as principal component analysis, could be applied. However, by reducing these signals to a single dimension, one loses potentially valuable information about how the features relate to one another at different time points (tumbling windows) and different scales (values of ϵ). If available, this information can be used to judiciously select the features salient to the task at hand while discarding those with redundant information. Thus, instead of treating these features as a one-dimensional vector, they are assembled into tensors that preserve the structural and temporal relationships inherent in the feature space.

3.6.1. Tensor formation

The waveform features for each patient are comprised of five groups: 6 HRV Taut String features, 6 ECG Taut String features, 152 DTCWPT wavelet features derived from the Taut String estimate of the ECG, 21 ABP features, and 21 SpO_2 features. Each feature is calculated for 5 tumbling windows and for 5 different values of ϵ per window. Prior to tensor formation the features within each tumbling window/value of ϵ are standardized, with the population means and standard deviations calculated for each feature/window/ϵ from the training set being used to standardize the corresponding feature/window/ϵ in the test set. Third-order tensors ($\epsilon \times$ feature $\times$ window) are then constructed for each of the five feature groups: the 6 HRV Taut String and 6 ECG Taut String features yield two $5 \times 6 \times 5$ tensors, the 152 DTCWPT wavelet features form a $5 \times 152 \times 5$ tensor, and the 21 ABP and 21 SpO_2 features each produce a $5 \times 21 \times 5$ tensor. An example of tensor formation for the ECG Taut String features is depicted in Fig. 3. Once formed, dimensionality reduction on these tensors then follows a two-step process: (1) a sequence of tensor analysis methods that determines the underlying structure within each tensor, and (2) extracting a reduced set of features.

3.6.2. Determining the tensor structure

The underlying structure of each tensor is determined by analyzing the tensors generated from the training set, as depicted in Fig. 4. The tensors resulting from each waveform feature group are stacked in the fourth mode to create a fourth-order tensor. For example, for the HRV Taut String features and a prediction interval of 30 min, this creates a tensor of size $5 \times 6 \times 5 \times M$, where M is the number of patients in the training set. The same stacking is done for the other groups of features.

Some of the groups (e.g., the DTCWPT ECG TS features) have too many features to make tensor decomposition feasible, thus Higher Order Singular Value Decomposition (HOSVD) [31–33] is used to reduce the number of features in those groups. HOSVD is a generalization of the singular value decomposition of a matrix. It can be also viewed as a higher order Principal Component Analysis (PCA) for data dimensionality reduction. HOSVD decomposes a tensor of order d into factor matrices $U_i (1 \leq i \leq d)$ and a core tensor. HOSVD can be used to reduce a dth order tensor of size $n_1 \times n_2 \times \ldots \times n_d$ to a core tensor of size $n_1' \times n_2' \times \ldots \times n_d'$ such that $n_i' \leq n_i$ for $1 \leq i \leq d$.

For each tensor X, HOSVD is applied to mode 2 (the feature mode), producing a version $\widehat{X}$. The relative error E is then calculated as

$$E = \frac{\left\| X - \widehat{X} \right\|}{\|X\|}, \tag{6}$$

with $\|X\|$ denoting the norm of the tensor X, defined as

$$\|X\| = \sqrt{\sum_{i_1, \ldots, i_d} (x_{i_1 \ldots i_d})^2}, \tag{7}$$

where $x_{i_1 \ldots i_d}$ is the $(i_1, i_2, \ldots, i_d)$th entry of X. If E is greater than 10%, HOSVD is repeated using a larger n_2' and terminates at the original dimension n_2 in mode 2 if no valid approximation is found. For example, the $5 \times 152 \times 5 \times M$ tensor of the DTCWPT ECG TS features is reduced to a $5 \times f_3 \times 5 \times M$ tensor, with $f_3 \ll 152$. The HOSVD algorithm also generates as output a matrix U_1 that gives the transformation from the 152-dimensional space to the f_3-dimensional space. This is repeated for the ABP and SpO_2 features, which results in tensors of size $5 \times f_4 \times 5 \times M$ and $5 \times f_5 \times 5 \times M$, as well as transformation matrices U_2 and U_3, respectively. The fourth-order tensors for each of the feature groups are then stacked in mode 2 (features) to obtain a $5 \times F \times 5 \times M$ tensor T, where $F = 6 + 6 + \sum_{i=3}^{5} f_i$.

In the final step of determining the tensor structure, a Canonical Polyadic (CP) decomposition [34] of the tensor T is computed. The CP decomposition factors a tensor into a sum of rank-1 tensors and can be considered another higher order generalization of singular value decomposition. In general, given a tensor $Y = \mathbb{R}^{n_1 \times \cdots \times n_d}$ and a rank r, its CP decomposition is the tensor

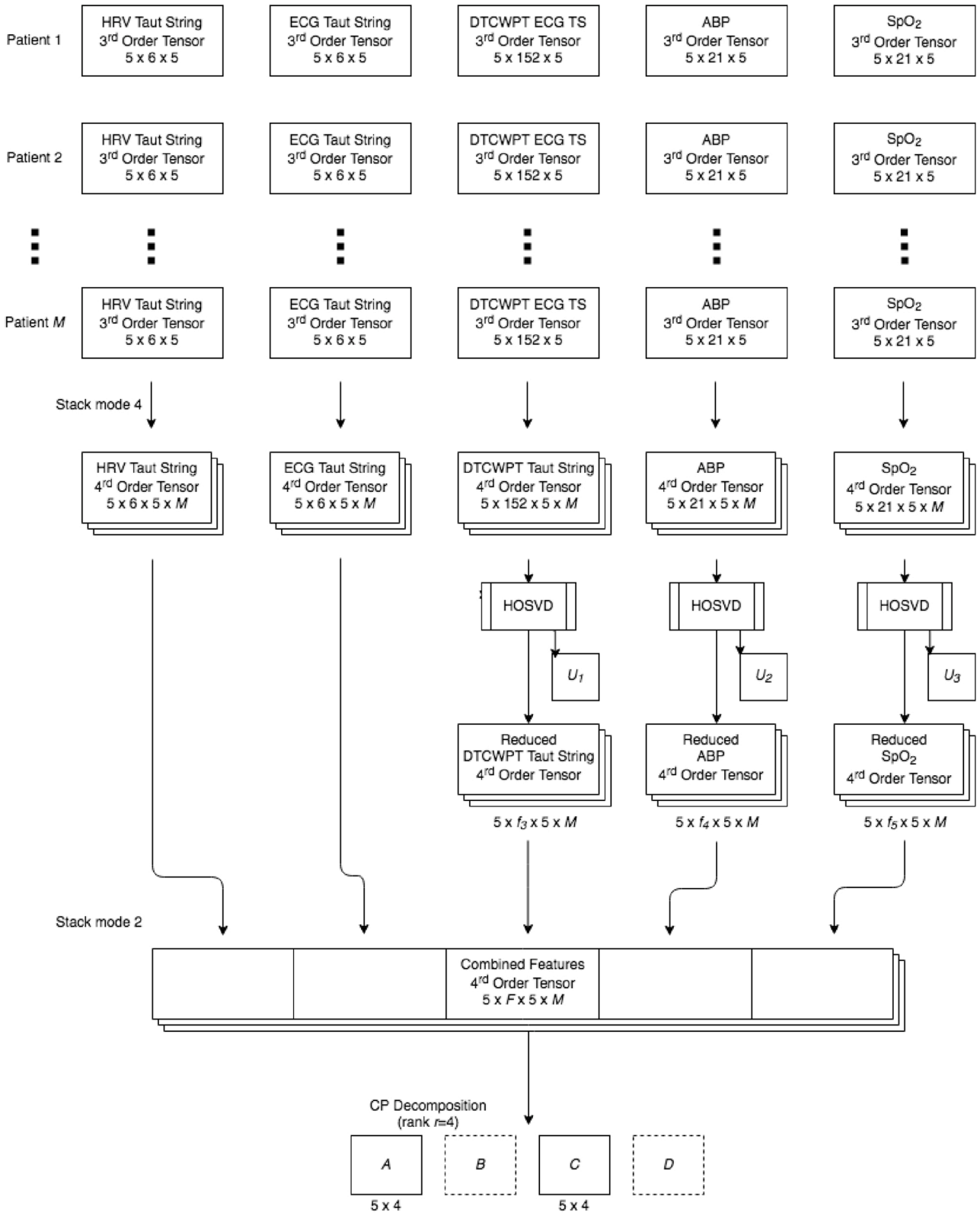

Fig. 4. A schematic diagram for the tensor-based feature reduction method. This process yields transformation matrices U_1, U_2, U_3 and factor matrices A and C that are subsequently used for feature extraction.

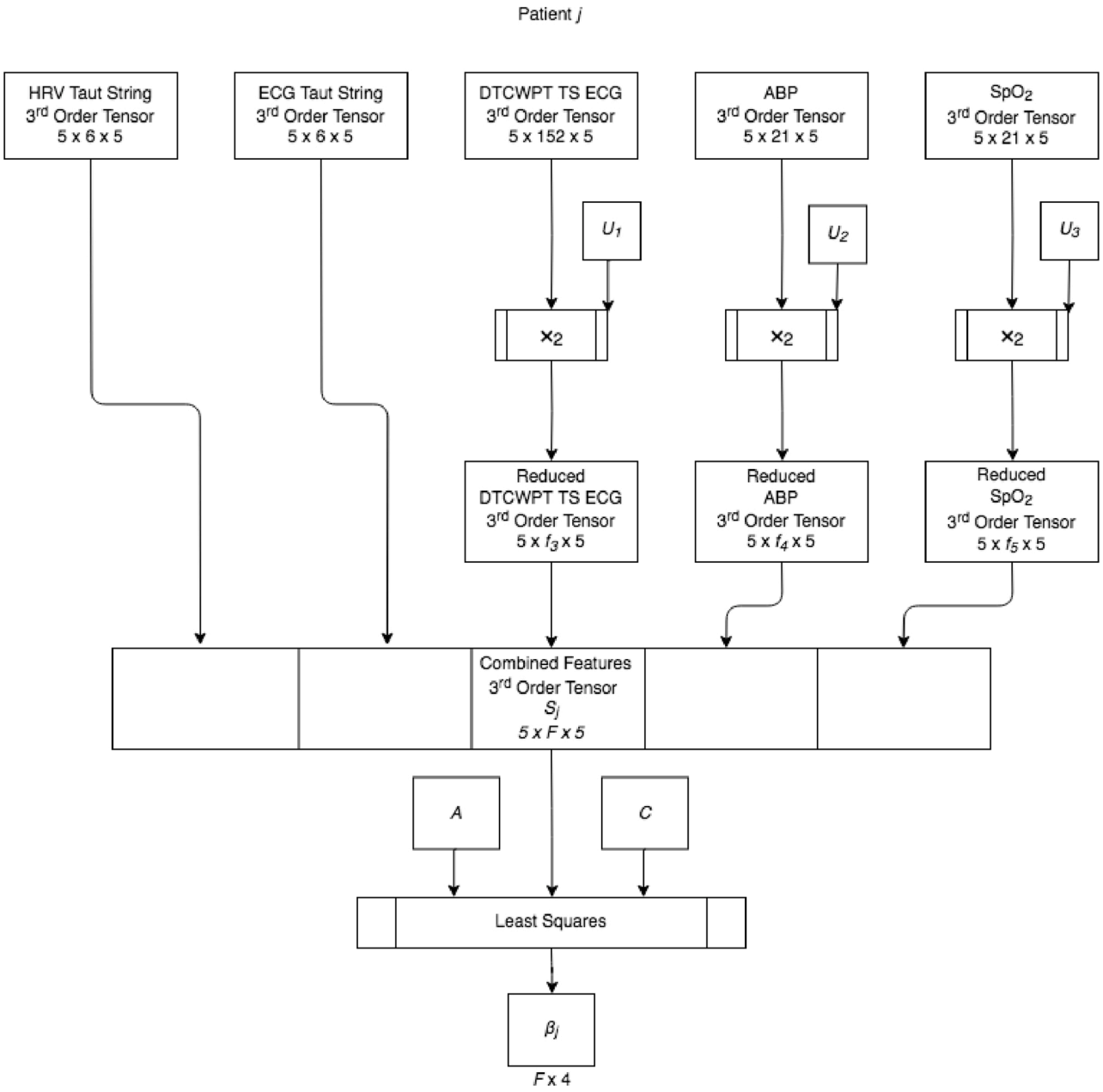

Fig. 5. Extracting features using the learned tensor structure. The transformation matrices and tensors from the previous step will be used to extract features from any patient. $\times_2$ denotes the mode-2 product of the tensor with the transformation matrix U_i.

$$\widehat{Y} = \sum_{i=1}^{r} v_1 \otimes \cdots \otimes v_d \tag{8}$$

such that $\|Y - \widehat{Y}\|$ is minimized. The smallest integer r for which such a decomposition exists is called the tensor rank of Y. As finding CP decompositions is NP-hard [35], heuristic methods are used instead. The Alternating Least Squares (ALS) method [34] is an iterative algorithm that attempts to find the best approximation $\widehat{Y}$ for a given rank r. The best approximation is usually chosen based on the highest fit, defined as fit $= 1 - E$, where E is the relative error as defined above. To determine the rank r, CP decomposition was applied to the feature tensors with increasing values of r starting from $r = 1$, with r being chosen as the minimum value where the fit of the resultant approximation T^* was greater than 0.9. In our experiments, $r = 4$ was the smallest value tested that consistently achieved the desired fit, thus this was the value used for all subsequent decompositions.

For each fourth-order tensor T used in our analysis, the ALS method is applied 100 times to determine the best approximation T^*, where

$$T^* = \sum_{i=1}^{r} a_i \otimes b_i \otimes c_i \otimes d_i. \tag{9}$$

The vectors $a_1, a_2, \ldots, a_r$ are grouped together to form the factor matrix A of size $5 \times r$. Similarly, a matrix C of size $5 \times r$ is created. The matrices B and D are discarded while the matrices A and C are retained for feature extraction.

3.6.3. *Extracting features from the reduced tensors*

For each patient in the training set and each group of features we have a third-order tensor as explained above. Using the transformations learned in the previous step, features can now be extracted for any patient in the dataset, including those in the test set. This process is depicted in Fig. 5. For any patient j the matrices U_1, U_2, U_3 are utilized to

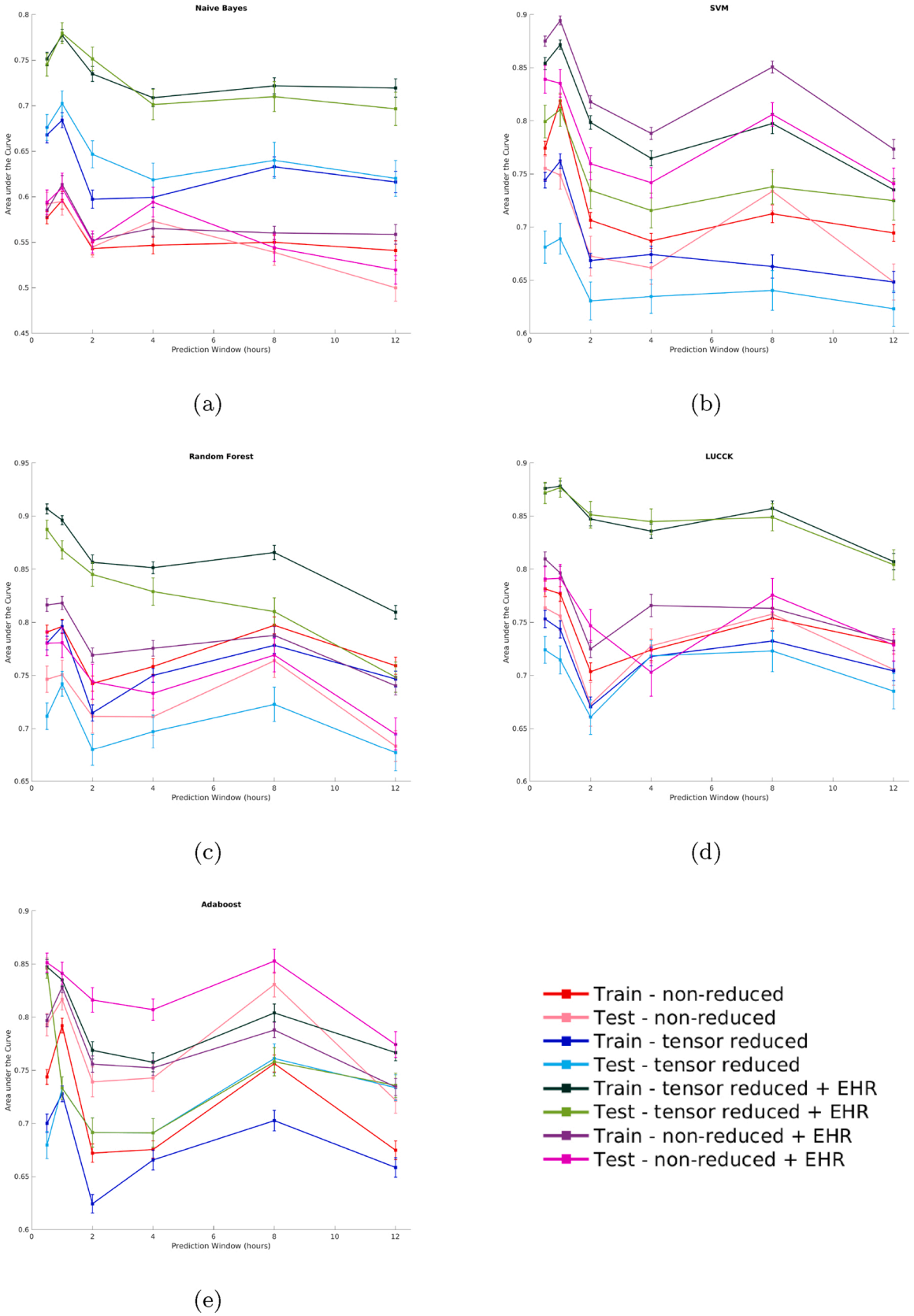

Fig. 6. The performance of (a) Naive Bayes (b) SVM (c) Random Forest, (d) LUCCK and (e) Adaboost models trained with non-reduced signal features, tensor-reduced signal features, the combination of non-reduced signal features with EHR features, and the combination of tensor-reduced signal features with EHR. Results on both the training and testing set are shown. The vertical bar at each data point indicates the 95% confidence interval obtained from the 101 cross-validations. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

reduce the number of DTCWPT ECG TS, ABP, and SpO_2 features. Next, the (dimension-reduced) tensors for all feature groups are stacked together to construct a $5 \times F \times 5$ tensor S_j. Note that this stacked tensor of combined features is third-order rather than fourth-order as in the previous step, as we are extracting features from each patient's data independently. The matrices A and C from the CP decomposition of T yield the vectors $a_1, a_2, \ldots, a_r$ and $c_1, c_2, \ldots, c_r$. We now find vectors $b_1, b_2, \ldots, b_r$ for which

$$\|S_j - \sum_{i=1}^{r} a_i \otimes b_i \otimes c_i\| \tag{10}$$

is minimal. This is a standard least squares problem that is easy to solve. The feature vectors $b_1, b_2, \ldots, b_r$ are concatenated to create a new feature matrix β_j of dimension $F \times r$.

On average, this tensor decomposition algorithm reduces the number of signal-based features from 5150 to 430.1386 (min–max = 412–448, SD = 8.5896), with $F = 107.5$ and $r = 4$, a 92% reduction.

3.7. Electronic health record feature reduction

As described in Section 3.5, 101 features were extracted from each patient's electronic health record. These features do not possess a structural or temporal relationship suitable for reduction using the above tensor-based method. However, there are many extant techniques that can be used for feature selection. Two such methods are principal component analysis (PCA) [36,37] and the minimum redundancy maximum relevance (MRMR) method [38,39]. During the development of this work both of these methods were tested for EHR feature reduction, but models utilizing the reduced EHR features consistently performed worse than the models for which non-reduction of the EHR features was performed. This is likely due to the majority of the EHR data being comprised of binary-valued features. Thus, in all subsequently reported results in this manuscript, the full set of 101 EHR features were utilized in modeling.

3.8. Machine learning

The physiological features were combined with all available EHR features and five types of machine learning models were trained. These include Random Forest [40], Naive Bayes [41], Support Vector Machine [42], Adaboost [43] and a recently developed kernel-based method called Learning Using Concave and Convex Kernels (LUCCK) [44]. All models were trained using the following combinations of signal-based and EHR features: (1) non-reduced waveform features (2) tensor-reduced waveform features, (3) non-reduced waveform features combined with all EHR features and (4) tensor-reduced waveform features combined with all EHR features.

A repeated three-fold cross-validation scheme was utilized for machine learning. The data were partitioned into three training folds plus a test fold. Each fold contained 25% of the samples and consisted of 35% positives and 65% negatives. No patient data occurred in more than one fold. Models were trained using the combined data of two folds, and an AUC score was calculated using the third fold for validation purposes. Model training was performed three times so that each fold was utilized exactly once to calculate an AUC score. The set of hyper-parameters achieving the highest AUC score averaged among the three validation scores was deemed optimal. Using these optimal hyper-parameters, a model was fit on the combined data from all three folds and subsequently evaluated using the test fold. Evaluation metrics obtained with the test set include AUC, F1 score, Precision, Recall, Sensitivity, and Specificity. The data were then shuffled, and the process was repeated for a total of 101 iterations.

3.8.1. Naive Bayes

A simple Naive Bayes (NB) model using a normal distribution and no hyperparameter tuning was trained as a baseline for model performance.

Table 3
The means and standard deviations of AUC for models trained with (a) all signal features, (b) tensor-reduced signal features, (c) all signal features plus EHR, and (d) tensor-reduced signal features plus EHR, for all prediction windows. The best performance at each prediction window is **bolded**, while the lowest standard deviation is *italicized*.

Prediction window (h)	Naive Bayes Mean (STD)	LUCCK Mean (STD)	RF Mean (STD)	SVM Mean (STD)	Adaboost Mean (STD)
(a) Mean AUC and standard deviation of non-reduced models, no EHR					
0.5	0.59 (0.07)	0.76 (0.06)	0.75 (0.06)	0.76 (0.06)	**0.79** *(0.05)*
1	0.59 (0.08)	0.76 (0.07)	0.75 (0.07)	0.75 (0.07)	**0.82** *(0.05)*
2	0.54 *(0.06)*	0.67 (0.11)	0.71 (0.08)	0.67 (0.10)	**0.74** (0.07)
4	0.57 *(0.08)*	0.73 (0.08)	0.71 (0.09)	0.66 *(0.08)*	**0.74** *(0.06)*
8	0.54 (0.07)	0.76 (0.08)	0.76 (0.08)	0.73 (0.09)	**0.83** *(0.06)*
12	0.50 (0.08)	0.71 (0.08)	0.68 (0.07)	0.65 (0.09)	**0.72** *(0.06)*
(b) Mean AUC and standard deviation of tensor-reduced models, no EHR					
0.5	0.68 (0.07)	**0.72** *(0.06)*	0.71 *(0.06)*	0.68 (0.08)	0.68 (0.07)
1	0.70 (0.07)	0.71 (0.07)	**0.74** (0.06)	0.69 (0.07)	0.73 *(0.05)*
2	0.65 (0.08)	0.66 (0.08)	0.68 *(0.08)*	0.63 (0.09)	**0.69** *(0.07)*
4	0.62 (0.09)	**0.72** *(0.07)*	0.70 (0.08)	0.63 (0.08)	0.69 *(0.07)*
8	0.64 (0.10)	0.72 (0.10)	0.72 (0.08)	0.64 (0.10)	**0.76** *(0.07)*
12	0.62 (0.10)	0.69 (0.09)	0.68 (0.09)	0.62 (0.08)	**0.73** *(0.06)*
(c) Mean AUC and standard deviation of non-reduced models with EHR					
0.5	0.59 (0.07)	0.79 (0.06)	0.78 (0.06)	0.84 (0.07)	**0.85** *(0.05)*
1	0.61 (0.08)	0.79 (0.06)	0.78 (0.07)	**0.84** (0.07)	**0.84** *(0.05)*
2	0.55 *(0.06)*	0.75 (0.08)	0.74 (0.08)	0.76 (0.08)	**0.82** *(0.06)*
4	0.59 (0.08)	0.70 (0.12)	0.73 (0.08)	0.74 (0.07)	**0.81** *(0.05)*
8	0.53 (0.08)	0.78 (0.08)	0.77 (0.08)	0.81 *(0.06)*	**0.85** *(0.06)*
12	0.52 (0.08)	0.73 (0.08)	0.69 (0.9)	0.74 (0.08)	**0.77** *(0.06)*
(d) Mean AUC and standard deviation of tensor-reduced models with EHR					
0.5	0.74 (0.06)	0.87 (0.05)	**0.89** *(0.04)*	0.80 (0.08)	0.85 (0.05)
1	0.78 (0.06)	**0.88** (0.05)	0.87 *(0.04)*	0.81 (0.08)	0.73 (0.05)
2	0.75 (0.07)	**0.85** *(0.06)*	**0.85** *(0.06)*	0.73 (0.09)	0.69 (0.07)
4	0.70 (0.09)	**0.84** *(0.06)*	0.83 (0.07)	0.72 (0.08)	0.69 (0.07)
8	0.71 (0.08)	**0.85** *(0.07)*	0.81 *(0.07)*	0.74 (0.08)	0.76 *(0.07)*
12	0.70 (0.09)	**0.80** (0.07)	0.75 *(0.07)*	0.72 (0.09)	0.74 *(0.06)*

3.8.2. Support vector machine

For the SVM models, linear, radial basis function (RBF), and 3rd-order polynomial kernels were evaluated. A grid search was performed to determine the best box constraint C and non-linear kernel scale γ, with logarithmically-spaced values chosen for $C \in [10^{-6}, 10^{13}]$ and $\gamma \in [10^{-12}, 10^{13}]$. Sequential minimal optimization (SMO) [45] was used for the optimization routine.

Table 4
The means and standard deviations of F1 scores for the final models trained with tensor-reduced signal features and EHR features for all prediction windows. The corresponding AUC results can be found in Table 3(d). The best performance at each prediction window is **bolded**, while the lowest standard deviation is *italicized*.

Prediction window (h)	Naive Bayes Mean (STD)	LUCCK Mean (STD)	RF Mean (STD)	SVM Mean (STD)	Adaboost Mean (STD)
0.5	0.68 (*0.06*)	0.77 (*0.06*)	**0.78** (*0.06*)	0.71 (0.07)	0.75 (0.06)
1	0.70 (*0.06*)	**0.77** (*0.06*)	0.76 (*0.06*)	0.71 (0.07)	0.65 (0.05)
2	0.69 (*0.06*)	**0.74** (0.07)	**0.74** (0.07)	0.65 (0.07)	0.62 (0.06)
4	0.65 (0.07)	**0.74** (0.07)	0.73 (0.07)	0.64 (0.07)	0.61 (0.06)
8	0.66 (*0.07*)	**0.75** (*0.07*)	0.72 (*0.07*)	0.63 (0.16)	0.68 (0.07)
12	0.66 (0.08)	**0.72** (0.08)	0.67 (*0.06*)	0.67 (0.08)	0.66 (0.06)

3.8.3. Random forest

The Random Forest method possesses a number of configurable parameters. These include the number of trees, minimum leaf size, node splitting criterion, number of predictors to sample, and the maximum number of decision splits for the decision trees. The values tested for the number of trees parameter were 50, 75, and 100; the minimum leaf size parameter values were 1, 5, 10, 15, and 20; the node splitting criteria tested were cross entropy and Gini impurity. The number of predictors to sample tested were $[10, 20, \ldots, 100]$; and the maximum number of decision splits parameter, expressed as a percentage of the training sample size, was 0.25, 0.50, 0.75, or 1.0. The optimal combination of these parameters was determined via a grid search approach.

3.8.4. Adaboost

Adaboost (adaptive boosting) is an ensemble machine learning algorithm. For each learning cycle, each weak learner is sequentially trained to classify the given input features into a binary outcome. The sum of the predictions made by these learners, weighted based on the error in the respective stages of training, is used to make the final prediction. In this study, 100 weak discriminant learners were trained via gradient boosting [46]. Grid search was employed to determine the best values of hyperparameters δ and γ, with $\delta \in [10^{-6}, 10^{3}]$ spaced logarithmically and $\gamma \in [0, 1]$. No regularization was used when training the models, i.e., the learning rate was set to 1.

3.8.5. Learning Using Concave and Convex Kernels (LUCCK)

The LUCCK method is a recently developed machine learning classifier [44]. It adjusts the concavity or convexity of similarity functions to determine independent models for each feature. This information is then used to adjust the importance of each feature for classification. The algorithm has two numerical hyper-parameters, λ and θ, which were optimized via grid search. In this study, approximately 50 values in the interval $[5 \times 10^{-5}, 3.0]$ were utilized for θ, while logarithmically-spaced values in the interval $[10^{-9}, 0.05]$ were evaluated as values of λ.

3.9. Comparison with other methods

In addition to testing multiple models and combinations of features, the proposed multimodal method was compared to one introduced in Melillo et al. [17]. Their method utilized common HRV features, such as correlation dimension, sample entropy, and absolute powers in various frequency bands, and employed a correlation feature selection (CFS) [47] algorithm on such signal features to determine the set of features with the lowest correlation. These features were then combined with EHR features to make predictions of adverse cardiovascular and cerebrovascular events. In this work, we utilize the Taut String HRV features (see 3.2) as well as the EHR features available in our dataset to emulate this method.

4. Experimental results

Plots of AUC vs. Prediction Window for the Naive Bayes, SVM, Random Forest, Adaboost, and LUCCK models are displayed in Fig. 6. Table 3 (a) compares the performance of models trained using the full set of signal-derived features, while Table 3 (b) shows the performance for models using the tensor-reduced set of signal features. Table 4 provides a performance comparison of the final models trained using both the tensor-reduced signal features and the selected EHR features. ROC curves for the final models for the half-hour prediction window are provided in Figure 7.

The Naive Bayes algorithm trained with the tensor-reduced signal features and EHR features achieved the highest AUC across all Naive Bayes models (Fig. 6(a): light green curve). The highest AUC was 0.78 for the 1-h prediction window. The lowest AUC was 0.697 for the 12-h window; however, the 95% confidence intervals for models trained on the 4, 8, and 12-h prediction windows overlapped with each other. With non-reduced signal features, the addition of EHR features did not improve the results significantly. When only signal features were used, the models trained with tensor-reduced signal features performed better than the model trained with non-reduced signal features consistently across all the prediction windows. However, overall, the Naive Bayes models performed worse than the SVM, Random Forest and LUCCK models.

The SVM algorithm, on the other hand, performed best when it was trained with the non-reduced signal features along with the EHR. The highest AUC across all SVM models was 0.84 for the 0.5-h prediction window trained with the non-reduced signals and EHR features (Fig. 6 (b): magenta curve). For the same set of features, the lowest AUC was 0.74 for the 4-h window; however, the 95% confidences interval for 2, 4, and 12-h prediction windows overlapped with each other. The SVM models trained on tensor-reduced signal features alone performed marginally worse than the SVM models trained using only the non-reduced signal features. The SVM models overall performed better than the Naive Bayes models, and performed consistently worse than the Adaboost models. With the exception of models with non-reduced signals and EHR features, SVM overall performed worse than the LUCCK and RF models.

The Random Forest models trained with the combination of tensor-reduced signal features and EHR features (Fig. 6(c): green curve) achieved the highest AUC scores across all prediction windows. For this model, the highest AUC was 0.89 for a half-hour prediction window. The lowest AUC was 0.75 and occurred for a 12-h prediction window. The AUC scores steadily decreased as the size of the prediction window increased, which is expected, as the information in the temporal data

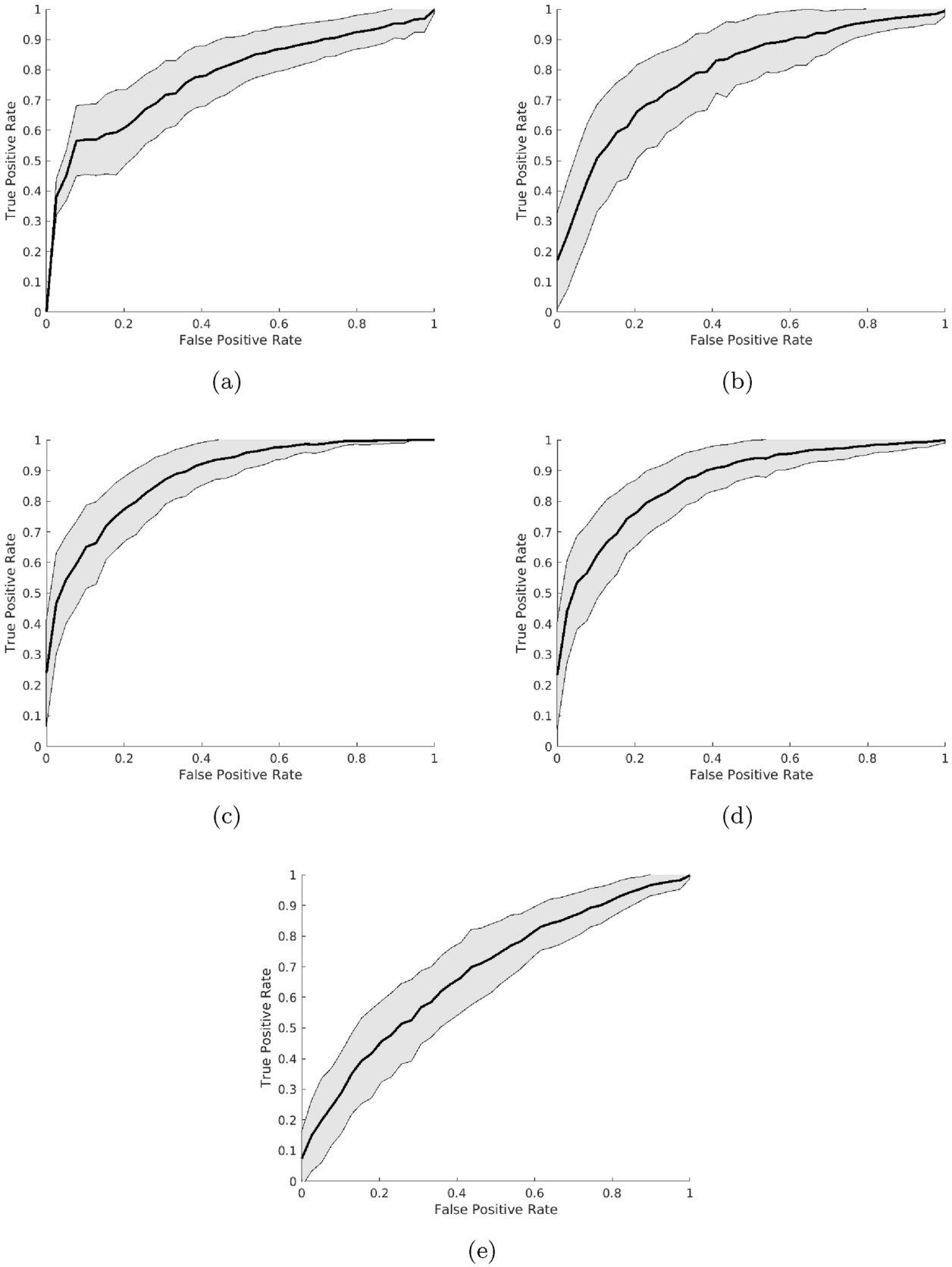

Fig. 7. ROC curves of the (a) Naive Bayes (b) SVM (c) Random Forest, (d) LUCCK, and (e) Adaboost models trained with the combination of tensor-reduced signal features and EHR for the half-hour prediction window. The thick line in the center represents the mean ROC curve over 101 cross-validations, while the shaded area represents one standard deviation from the mean.

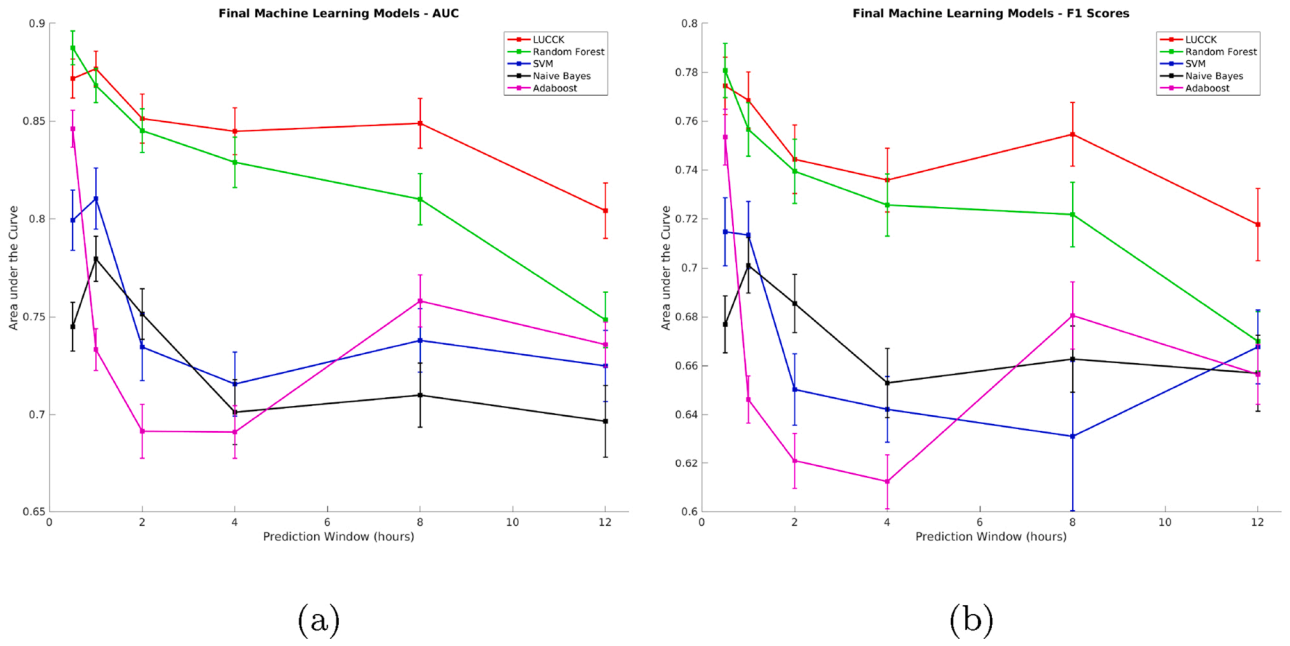

Fig. 8. (a) AUC and (b) F1 scores of the final tuned LUCCK, Random Forest, SVM, and Naive Bayes models trained with tensor-reduced signal features and EHR features. The vertical bar at each data point indicates the 95% confidence interval obtained from the iterations.

should become less relevant as the time between their measurement and the event increases. The AUC for the random forest models trained with non-reduced signal features (Fig. 6(c): pink curve) performed comparably to those trained with reduced signal features (Fig. 6(c): cyan curve) for prediction windows of 1, 2, 4, and 12 h. The AUC for the random forest models trained on non-reduced signal features was greater than the AUC for random forests trained on tensor-reduced signal features for prediction windows of a half-hour and 8 h. However, when EHR features are included, this trend is reversed – the tensor-reduced signal features significantly outperform the non-reduced features across all windows. For the models trained with tensor-reduced signal features and the model trained with non-reduced signal features, there is a noticeable drop in AUC for a 2-h prediction window, followed by an increase that leads to a noticeable peak in performance with an 8-h prediction window. For the model trained with tensor-reduced signal features (Fig. 6(c): cyan curve), the AUC score at the 1-h prediction window is better than the AUC score for a half-hour prediction window.

The LUCCK models trained with the combination of tensor-reduced signal features and EHR features (Fig. 6(d): green curve) outperformed the LUCCK model trained on tensor-reduced signal features (Fig. 6(d): cyan curve) and the LUCCK model trained on the non-reduced signal features (Fig. 6(d): pink curve). For this best model, the highest AUC score of 0.87 was achieved at prediction windows of 0.5 h and 1 h, respectively. These two scores are not statistically different. The AUC values of 0.85, 0.84, and 0.85 for prediction windows of 2, 4, and 8 h, respectively, are not statistically significantly different from each other. The lowest AUC was achieved with a prediction window of 12 h. The LUCCK models trained with non-reduced signal features achieve AUC scores that are comparable to the corresponding model trained on tensor-reduced signal features for prediction windows of 2, 4, 8, and 12 h. The LUCCK models trained with non-reduced signal features achieve AUC scores that are statistically significantly better than the corresponding AUC scores of the models trained on tensor-reduced signal features for prediction windows of 0.5 h and 1 h. When EHR features are added to the non-reduced signal features, the performance improves slightly for most windows but there is a noticeable drop for the 2- and 4-h windows. For the models trained with tensor-reduced signal features and the models trained with non-reduced signal features (Fig. 6(d)), the minimum AUC scores occur with a 2-h prediction window, followed by an increase in performance that peaks at the 8-h prediction window. The AUC scores at these 8-h markers are statistically significantly different from the AUC obtained with the 2-h prediction window.

The Adaboost models performed better when the signals were not reduced: the best AUC was achieved for the model trained with non-reduced signals and EHR features during the half-hour window (Fig. 6). In models with non-reduced signal features, with or without EHR, Adaboost was the best performing model overall, whereas its performance was worse with the tensor-reduced signal features. Across all feature sets, there was a noticeable peak in the 8-h window with the Adaboost models, a trend also observed in some SVM, RF and LUCCK models.

4.1. *Advantage of tensor-based dimensionality reduction*

From Fig. 6(a) (blue/cyan and red/pink curves) it can be observed that the tensor-based dimensionality reduction method leads to increased model performance for Naive Bayes, Random Forest, and LUCCK, especially when EHR features are included. For the Naive Bayes model, the tensor reduction method yields an average increase of 0.09 when compared to the model trained on all signal-based features. With the EHR features included, the tensor reduction yielded an even more dramatic improvement of 0.16 in average AUC compared to the models with non-reduced signal features. For both LUCCK and RF models with EHR included, the tensor reduction yielded an average increase in the AUCs of 0.09 each. Although this trend did not hold for the SVM and Adaboost models, the improvement achieved by tensor reduction on the Naive Bayes, LUCCK, and RF was larger than the reduction observed in SVM and Adaboost model performance. An advantage of this dimensionality reduction method is that it transforms the signal-based features from a set of 5150 into a much smaller set consisting of 430 features on average (see Section 3.6.3 for further details). This 92% reduction in the number of features results in a much simpler model that achieves statistically significant higher AUC values for each prediction window.

4.2. *Incorporation of EHR data into modeling process*

For each type of machine learning algorithm in this study, the models

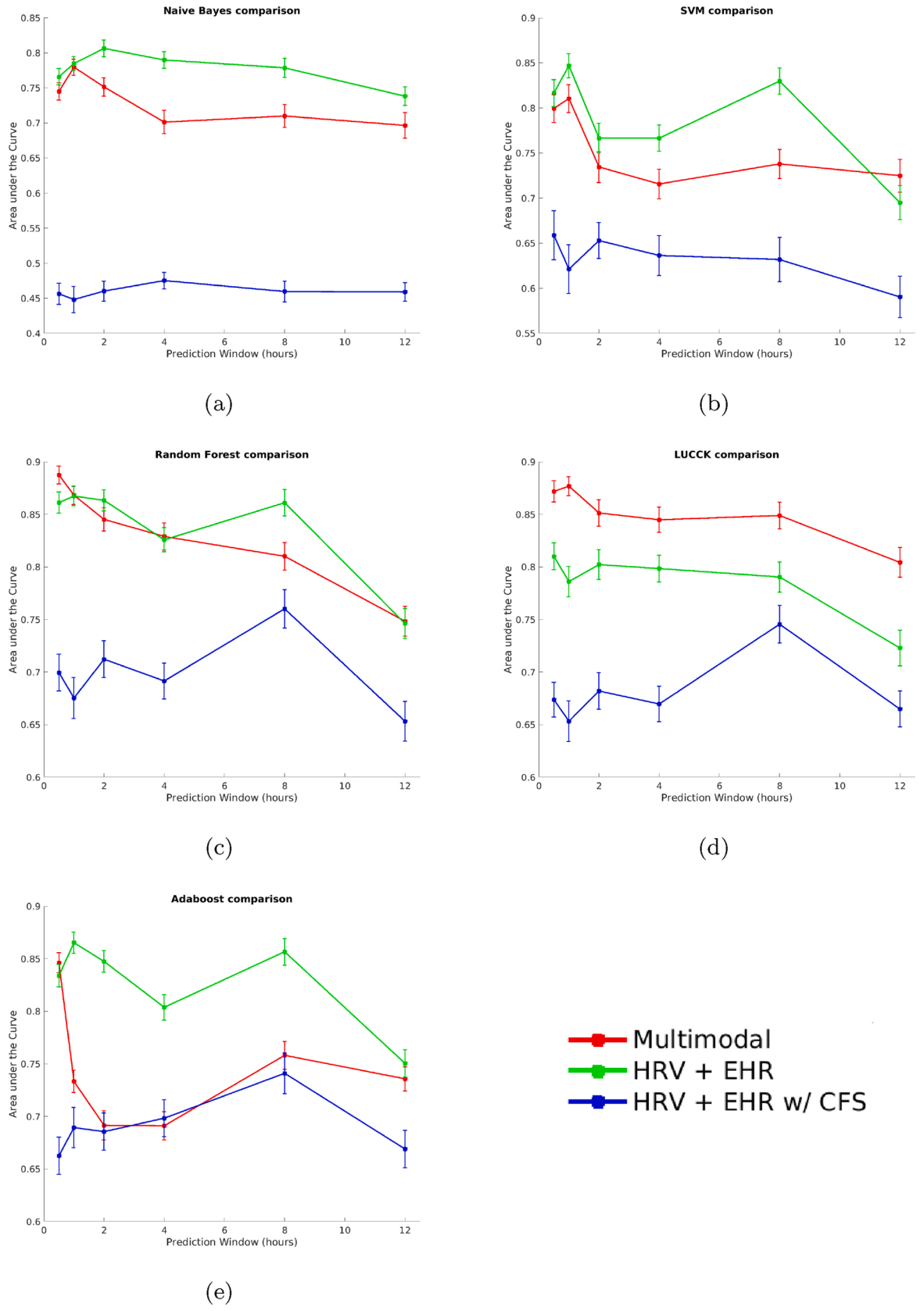

Fig. 9. (a) Naive Bayes (b) SVM (c) Random Forest, (d) LUCCK and (e) Adaboost models trained according to the multimodal pipeline proposed in this manuscript and a pipeline proposed by Melillo et al. [17] with and without CFS. The vertical bar at each data point indicates the 95% confidence interval obtained from the 101 cross-validations.

trained using the combination of tensor-reduced signal features and EHR features consistently outperformed models trained on the tensor-reduced signal features and models trained on the non-reduced signal features.

For Naive Bayes, the incorporation of EHR data into the model yielded an increase in test set AUC by an average of 0.08 when compared to the model trained using reduced signal features alone, and an average increase of 0.17 when compared to the model trained only using non-reduced signal features.

Random forest achieved an average increase of 0.10 in test set AUC through the combination of EHR and reduced tensor features when compared to the model trained on non-reduced signal features alone, and an average increase of 0.13 when compared to the random forest model trained only on the reduced signal features.

LUCCK achieved an average increase of 0.12 in test AUC through the combination of EHR and reduced tensor features when compared to the LUCCK model trained only using non-reduced signal features, and an average increase of 0.15 when compared to the LUCCK model trained only using the reduced signal features.

SVM achieved a 0.05 average increase in test AUC through incorporation of EHR data when compared to the SVM model trained on non-reduced signal features alone, and an average increase of 0.10 when compared to the SVM model trained only with reduced signal features.

Adaboost achieved a 0.05 average increase in test AUC by incorporating EHR data when using non-reduced signal features and an average increase of 0.03 when using tensor-reduced signal features.

From Fig. 8(a) it can be seen that the LUCCK model performed best, random forest performed second best, followed by the SVM model, and then the Naive Bayes model. The LUCCK and random forest models achieved higher AUC scores than SVM and Naive Bayes for all prediction windows. The LUCCK and random forest algorithms achieved comparable performance for prediction windows of 0.5, 1, 2, and 4 h. However, for prediction windows of 8 and 12 h, the LUCCK model achieves a higher AUC than random forest. The SVM and Naive Bayes models achieved comparable performance for all prediction windows greater than 1.0 h. For the half-hour and 1-h prediction windows, SVM achieved higher AUC than Naive Bayes.

Table 4 contains a summary of F1 scores for all models. LUCCK and random forest models achieve higher F1 scores than SVM and Naive Bayes models for all prediction windows (Fig. 8(b)). Furthermore, LUCCK and Random Forest achieve comparable F1 scores for prediction windows of 0.5, 1, 2, and 4 h; for 8 and 12 h, LUCCK achieves higher F1 scores than random forest. SVM and Naive Bayes achieve comparable F1 scores for prediction windows of 1, 4, 8, and 12 h. SVM achieved statistically higher F1 score than Naive Bayes for a prediction window of a half-hour; however, for a prediction window of 2 h, Naive Bayes achieves a statistically higher F1 score than SVM.

4.3. Comparison with other methods

A comparison of the performance of the proposed multimodal method to that proposed by Melillo et al. [17] when HRV and EHR features is depicted in Fig. 9. When compared with the proposed method without their feature selection method, CFS, the multimodal approach performed better with the LUCCK models, comparably with RF and worse with the Naive Bayes, Adaboost, and SVM models. However, when CFS was used for feature selection, the performance was consistently worse across all models.

5. Discussion

Naive Bayes, RF, and LUCCK machine learning methods consistently achieved their respective highest AUC scores when utilizing the combination of tensor-reduced signals and EHR features. On the other hand, Adaboost and SVM achieved better performance when non-reduced signal features were used along with the EHR. No matter whether tensor-reduced or non-reduced signals were used, the addition of EHR improved model performance. This is not surprising, as EHR features include clinical interventions, such as the administration of cardiovascular infusions or medications, which are directly related to patient recovery and outcomes. The inclusion of such information into a real-time clinical decision support system would be invaluable in providing comprehensive monitoring of patients.

Among all models, LUCCK models trained with tensor-reduced signals features and EHR performed best, achieving AUC scores equal to or greater than those achieved by the other models. The random forest models trained on tensor-reduced signals and EHR features was the second best. These two top performing models were nearly comparable in performance for prediction windows up to 4 h, after which random forest fared worse by comparison. An explanation for their ability to outperform the SVM and Naive Bayes models for all prediction windows is that LUCCK and random forest are equipped with internal mechanisms that weight or select the most informative features. In this study, the reduced feature set contained approximately 530 values, about three times the total number of samples available for training. In this situation, additional feature selection is typically beneficial to the predictive modeling process, particularly in helping to avoid overfitting. The modeling results for the Naive Bayes algorithm (Fig. 6(a)) demonstrate that tensor-based dimensionality reduction can be useful for training machine learning models that lack an internal dimensionality reduction or feature selection process similar to random forest and LUCCK.

For LUCCK, random forest, and the SVM models AUC scores generally decreased as the size of the prediction window increased (Fig. 8(a)). In some cases, such as for prediction windows of 2, 4, and 8 h, LUCCK performed comparably well just before suffering a performance decrease at 12 h. This general trend is not unexpected since the information captured during the analysis window of fixed duration should be more predictive of events occurring in the short-term and less predictive of events occurring further into the future. With the exception of the increase in AUC from 0.5 h to 1 h, this trend is true of the Naive Bayes and Adaboost models as well.

To explore mechanisms potentially explaining the increase in AUC when prediction windows were expanded from 0.5 h to 1 h, we examined (i) relative ratios of each of the nine adverse event types contributing to the model, across all prediction window sizes (as longer windows precluded analysis of early adverse events) and (ii) granularity of model features. Given that (i) inspection of relative ratios of adverse event types analyzed across varying prediction windows yielded no discernible trend and (ii) all model features were documented to the nearest minute, the observed increase in model performance from 0.5 h to 1 h warrants further study and our findings within this prediction window range should be interpreted with caution.

Reducing the tensor data did not necessarily lead to an increase in AUC when compared to modeling with the non-reduced tensor features (Table 3). For example, the LUCCK model trained with the non-reduced tensor data outperformed the model trained on the reduced tensor data for prediction windows of 0.5 and 1 h. It is not surprising that tensor dimension reduction occasionally yields a slightly reduced performance,

since the dimension reduction process discards information, some of which is potentially useful, in transforming the tensor features from a set of 5150 to a smaller set of 430 on average. However, the addition of EHR data into the tensor-reduced signal feature set mitigated this problem, mostly outperforming the models trained with non-reduced signals and EHR features.

The comparison to the method introduced in Melillo et al. [17] shows that the multimodal approach with tensor-reduced signals performs better than any models reduced using CFS, although they perform generally worse than the non-reduced HRV features with the exception of the LUCCK models. This suggests that the LUCCK is likely the best model to learn from the tensor-reduced signal information, whereas other machine learning models suffer from the loss of information resulting from tensor reduction.

A challenge in utilizing current methods for tensor decompositions such as CP-ALS is that the resulting approximation to the tensor may change based on the chosen CP-ALS input parameters. We have attempted to rectify this by running CP-ALS 100 times during the training process to find the best fit. Though the approximation to the CP decomposition as determined by CP-ALS may not be unique, once the training process is complete the resulting factor matrices are used to transform testing samples through a deterministic process (up to numerical precision) that only utilizes matrix computations. Moreover, by repeating the experiments 101 times across as many shuffles of the dataset, we observe that we can achieve high model performance and low standard deviation.

6. Conclusion

In this study a multimodal approach that incorporates salient physiological signals and EHR data was proposed to predict the onset of hemodynamic decompensation. Advanced signal processing techniques such as Taut String estimation and Dual-tree Complex Wavelet Packet Transform were employed to extract complex and non-obvious patterns from ECG signals, while a novel tensor-based dimensionality reduction method was used to reduce the size of the resultant feature space. Various models were tested at multiple prediction windows, with the random forest model achieving an AUC of 0.89 in predicting adverse events a half-hour in advance, and the LUCCK model an AUC of 0.80 12 h in advance.

Overall, the analyses show that the information from continuous signals collected in the intensive care unit, combined with EHR data, can be utilized in the clinical decision support system for predicting adverse events several hours in advance, utilizing features that are not readily available for clinical interpretation. Such a system would allow for early interventions and could potentially lead to better patient outcomes in the critical care setting.

Conflict of interest statement

NT, HD, KN, and JG: A patent application related to the tensor-based algorithms described in this manuscript has been filed through the University of Michigan's Office of Technology Transfer.

All other authors declare no conflict.

Acknowledgment

This work was partially supported by the U.S. Army Medical Research and Material Command Program under Grant No. W81XWH-17-2-0012 and by the National Science Foundation under Grant No. 1837985.

Appendix A. Adverse events

Tables 5 and 6 below detail the number of positive patients and adverse events available at each prediction interval:

Table 5
The number of patients having an adverse event at each prediction interval length.

Event/Interval (h)	0.5	1	2	4	8	12
Low cardiac index	24	26	25	26	25	29
Sustained low MAP	4	4	4	4	4	2
Mortality	6	5	4	2	2	1
Epinephrine bolus	5	5	5	4	4	4
New inotrope	7	7	7	4	4	4
Inotrope escalated	7	7	7	4	2	1
New vasopressor	4	4	2	1	2	2
New vasopressor	14	14	13	11	6	5
Vasopressor escalated	0	1	0	1	0	1
Total	71	73	67	57	49	49

Table 6
The total number of adverse events that occurred across all patients at various prediction interval lengths.

Event/interval (h)	0.5	1	2	4	8	12
Low cardiac index	31	32	31	31	28	31
Sustained low MAP	4	4	4	4	4	2
Mortality	6	5	4	2	2	1
Epinephrine bolus	6	6	6	5	5	4
New inotrope	8	8	8	5	5	5
Inotrope escalated	7	7	7	4	2	1
New vasopressor	4	4	2	1	2	2
New vasopressor	19	19	16	14	8	6
Vasopressor escalated	0	1	0	1	0	1
Total	85	86	78	67	56	53

Appendix B. QRS complex detection method

This appendix describes the peak detection method used in Section 3.2 to determine the QRS complexes within an ECG signal. The method consists of two steps: (1) filtering the signal to enhance the prominence of QRS complexes and (2) detecting the R-peaks. These steps are described in the following sections.

B.1 Description of the `PeakFilter` *algorithm*

The function `PeakFilter` takes an ECG signal $f(t)$ as input and gives a function $f_{\texttt{peak}}(t)$ with values in the interval $[0, 1]$ as output. It is used by the peak detection algorithm `PeakDetection` discussed later. The `PeakFilter` function is designed so that the value of $f_{\texttt{peak}}(t)$ is 1 if $f(t)$ has a peak that is similar to a QRS complex. Below is the pseudocode for `PeakFilter`:

Algorithm 1. Peak Filtering

1:	**function** `PeakFilter`$f(t)$
2:	$f_1(t) = \sqrt{max\begin{cases}(f(t)-f(t-w_{\text{peak}}))\cdot\\(f(t)-f(t+w_{\text{peak}})),\\0\end{cases}}$
3:	$f_{\text{smooth}}(t) = (f_1 \star p)(t)$
4:	$f_{\text{norm}}(t) = f_1(t)/f_{\text{smooth}}(t)$
5:	$f_{\text{peak}}(t) = \Phi(f_{\text{norm}}(t))$
6:	**return** $f_{\text{peak}}(t)$
7:	**end function**

In line 2 of Algorithm 1, a nonlinear filter is applied to $f(t)$ to obtain $f_1(t)$. This filter enhances the QRS complex. The function $f_1(t)$ will be nonzero if and only if either $f(t)$ is larger than $f(t - w_{\text{peak}})$ and $f(t + w_{\text{peak}})$ (a positive peak) or $f(t)$ is smaller than $f(t - w_{\text{peak}})$ and $f(t + w_{\text{peak}})$ (a negative peak). The value of $2{\cdot}w_{\text{peak}}$ is approximately the width of the QRS complex. The normal width of a QRS complex is in the range of 70 and 100 ms. Consequently, the value $w_{\text{peak}} = 0.035$ s (and thus $2w_{\text{peak}} = 70$ ms) was chosen.

In line 3, $f_1(t)$ is smoothed by convolving it with the function $p(t)$, depicted in Fig. 10. The linear filter used is defined by $p(t) = (u \star u \star u)(t)$, where $u(t)$ is the function that is equal to $1/w_{\text{window}}$ on the interval window $[-\frac{1}{2}w_{\text{window}}, \frac{1}{2}w_{\text{window}}]$ and equal to 0 outside this interval. The window width chosen for this analysis was $w_{\text{window}} = 1$ s, but this parameter could be varied.

Instead of $p(t)$ one could also use another nonnegative convolutional filter that is normalized such that $\int p(t)\,\text{dt} = 1$.

In line 4 we normalize $f_1(t)$ to get $f_{\text{norm}} = f_1(t)/f_{\text{smooth}}$. The normalization is local, so that the function f_{normal} is not affected by change of the amplitude of an ECG signal over time.

In line 5 the thresholding function $\Phi(s)$ (depicted in Fig. 11 above) is applied to the normalized signal f_{norm} to obtain the output $f_{\text{peak}}(t)$. The thresholding function used is a piecewise linear function defined by:

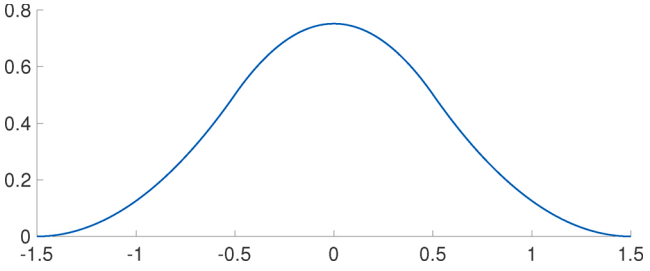

Fig. 10. The linear filter function $p(t)$.

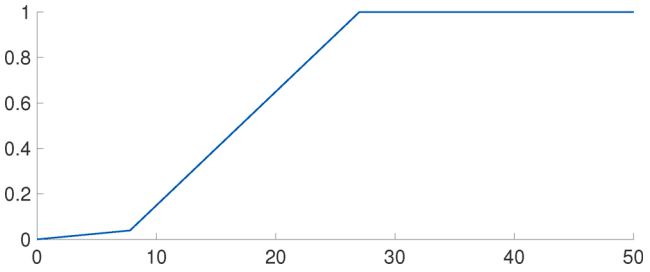

Fig. 11. The threshold function $\Phi(s)$.

$$\Phi(s) = \begin{cases} 0 & \text{if } s \le 0; \\ s/b & \text{if } 0 \le s \le b(c-a)/(b-a); \\ (s+a-c)/a & \text{if } b(c-a)/(b-a) \le s \le c; \\ 1 & \text{if } s \ge c \end{cases}, \tag{11}$$

where $0 < a < c < b$. The parameters used were $a = 20$, $b = 200$ and $c = 27$. The threshold c can be interpreted as indicating a definitive peak near time t if $f_{\text{norm}}(t) \ge c$, with certainty decreasing for smaller values of $f_{\text{norm}}(t)$. Other threshold functions $\Phi(s)$ could also be used.

B.2 Description of the `PeakDetection` *algorithm*

Calling the function `PeakDetection`$(f(t), t_0, t_{\max})$ finds all the R-peaks of the ECG signal $f(t)$ in the interval $(t_0, t_{\max})$. The function outputs the timestamps $t_1, t_2, \ldots, t_i$ where R-peaks are located. The time interval between two peaks is assumed to be in the interval $[s_{\min}, s_{\max}]$; this means that the heart rate is assumed to lie in the interval $[60/s_{\max}, 60/s_{\min}]$. The values $s_{\min} = .25$ s and $s_{\max} = 4$ s were used in this study, i.e., the heart rate is between 15 and 240 BPM.

Algorithm 2. Peak Detection

1:	**function** `PeakDetection`$f(t), t_0, t_{\max}, s_{\min}, s_{\max}$
2:	$f_{\text{smooth}}(t) = 50 \int_{t-.01}^{t+.01} f(t)\,\text{dt}$
3:	$f_{\text{peak}}(t) =$ `PeakFilter`$(f_{\text{smooth}}(t))$
4:	$i = -1$
5:	**repeat**
6:	$i = i + 1$
7:	Choose $s \in [s_{\min}, s_{\max}]$ such that $f(t_i + s)\alpha(s)$ is maximal
8:	$t_{i+1} = t_i + s$
9:	**until** $t_{i+1} \ge t_{\max}$
10:	**return** $t_1, t_2, \ldots, t_i$
11:	**end function**

In line 2, the ECG signal $f(t)$ is smoothed slightly using a moving average over window of length 0.02s to obtain f_{smooth}. The `PeakFilter` function (Algorithm 1) is applied to $f_{\text{smooth}}(t)$ to obtain the peak signal $f_{\text{peak}}(t)$ in line 3.

In lines 7 and 8, $t_{i+1} = t_i + s$ is defined, where $s \in [s_{\min}, s_{\max}]$ is chosen such that $f(t_i + s)\alpha(s)$ is maximal. The function $\alpha(s)$ is called the anticipation function and influences which of the nearby peaks will be labeled as the next R-peak by the algorithm. The anticipation function used was

$$\alpha(t) = \frac{t^2}{1 + 100(t - 0.2)^4} \tag{12}$$

for $t \in [s_{\min}, s_{\max}]$, and is depicted in Fig. 12 below:

The choice of $\alpha(s)$ in this study expresses that the interval between two consecutive peaks is more likely to be 0.5 s than 2 s. If $f(t)$ has peaks of similar height at $t_i + .5$ and $t_i + 2$, then $f(t_i + 0.5)\alpha(0.5) > f(t_i + 2)\alpha(2)$ (because $\alpha(0.5) > \alpha(2)$) and $t_i + 0.5$ is considered more likely to be the next peak than $t_i + 2$. However, if the peak at $t_i + 2$ is much larger than the peak $t_i + 0.5$ then $f(t_i + 0.5)\alpha(0.5) < f(t_i + 2)\alpha(2)$ and $t_i + 2$ is considered to be more likely as the next peak than $t_i + 0.5$.

In lines 7 and 8, i runs through successive integer values starting from 0. In the last iteration, i is minimal such that $t_{i+1} > t_{\max}$ and $t_1, t_2, \ldots, t_i$ are all the peaks in the open interval $(t_0, t_{\max})$.

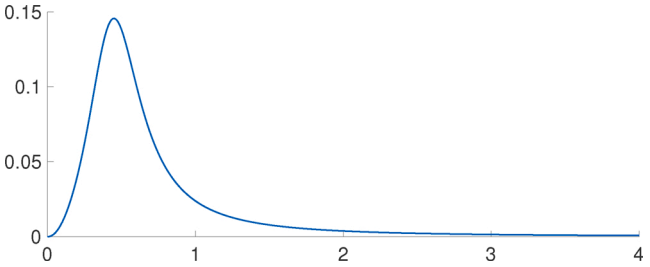

Fig. 12. The anticipation function $\alpha(s)$.

Appendix C. Supplementary data

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.artmed.2021.102032.

References

[1] O'Brien SM, Feng L, He X, Xian Y, Jacobs JP, Badhwar V, et al. The society of thoracic surgeons 2018 adult cardiac surgery risk models: part 2-statistical methods and results. Ann Thorac Surg 2018;105(5):1419–28.

[2] Vandendriessche B, Abas M, Dick TE, Loparo KA, Jacono FJ. A framework for patient state tracking by classifying multiscalar physiologic waveform features. IEEE Trans Biomed Eng 2017;64(12):2890–900.

[3] Tsien CL, Fackler JC. Poor prognosis for existing monitors in the intensive care unit. Crit Care Med 1997;25(4):614–9.

[4] Chambrin M-C, Ravaux P, Calvelo-Aros D, Jaborska A, Chopin C, Boniface B. Multicentric study of monitoring alarms in the adult intensive care unit (icu): a descriptive analysis. Intensive Care Med 1999;25(12):1360–6.

[5] Imhoff M, Kuhls S. Alarm algorithms in critical care monitoring. Anesth Analg 2006;102(5):1525–37.

[6] Hagenouw RR. Should we be alarmed by our alarms? Curr Opin Anesthesiol 2007; 20(6):590–4.

[7] Johnson KR, Hagadorn JI, Sink DW. Alarm safety and alarm fatigue. Clin Perinatol 2017;44(3):713–28.

[8] Petersen EM, Costanzo CL. Assessment of clinical alarms influencing nurses' perceptions of alarm fatigue. Dimens Crit Care Nurs 2017;36(1):36–44.

[9] Belle A, Thiagarajan R, Soroushmehr S, Navidi F, Beard DA, Najarian K. Big data analytics in healthcare. BioMed Res Int 2015;2015.

[10] Ismail F, Davies M. Integrated monitoring and analysis for early warning of patient deterioration. Br J Anaesth 2007;98(1):149–50.

[11] Javed F, Savkin AV, Chan GS, Mackie JD. Modeling and model predictive control of hemodynamic variables during hemodialysis. In: 49th IEEE conference on decision and control (CDC); 2010. p. 4673–8.

[12] Koch E, Lovett S, Nghiem T, Riggs RA, Rech MA. Shock index in the emergency department: utility and limitations. Open Access Emerg Med: OAEM 2019;11:179.

[13] Le TQ, Bukkapatnam ST, Komanduri R. Real-time lumped parameter modeling of cardiovascular dynamics using electrocardiogram signals: toward virtual cardiovascular instruments. IEEE Trans Biomed Eng 2013;60(8):2350–60.

[14] Kunhimangalam R, Ovallath S, Joseph PK. A clinical decision support system with an integrated emr for diagnosis of peripheral neuropathy. J Med Syst 2014;38(4): 38.

[15] Convertino VA, Moulton SL, Grudic GZ, Rickards CA, Hinojosa-Laborde C, Gerhardt RT, et al. Use of advanced machine-learning techniques for noninvasive monitoring of hemorrhage. J Trauma Acute Care Surg 2011;71(1):S25–32.

[16] Potes C, Conroy B, Xu-Wilson M, Newth C, Inwald D, Frassica J. A clinical prediction model to identify patients at high risk of hemodynamic instability in the pediatric intensive care unit. Crit Care 2017;21(1):282.

[17] Melillo P, Izzo R, Orrico A, Scala P, Attanasio M, Mirra M, et al. Automatic prediction of cardiovascular and cerebrovascular events using heart rate variability analysis. PLOS ONE 2015;10(3):e0118504.

[18] Salomão Jr E, Otsuki DA, Correa AL, Fantoni DT, dos Santos F, Irigoyen MC, et al. Heart rate variability analysis in an experimental model of hemorrhagic shock and resuscitation in pigs. PLOS ONE 2015;10(8).

[19] Lan K-c, Raknim P, Kao W-F, Huang J-H. Toward hypertension prediction based on ppg-derived hrv signals: a feasibility study. J Med Syst 2018;42(6):103.

[20] Sessa F, Anna V, Messina G, Cibelli G, Monda V, Marsala G, et al. Heart rate variability as predictive factor for sudden cardiac death. Aging (Albany NY) 2018; 10(2):166.

[21] Koko KR, McCauley BD, Gaughan JP, Fromer MW, Nolan RS, Hagaman AL, et al. Spectral analysis of heart rate variability predicts mortality and instability from vascular injury. J Surg Res 2018;224:64–71.

[22] Chiew CJ, Liu N, Tagami T, Wong TH, Koh ZX, Ong ME. Heart rate variability based machine learning models for risk prediction of suspected sepsis patients in the emergency department. Medicine 2019;98(6).

[23] Swor DE, Thomas LF, Maas MB, Grimaldi D, Manno EM, Sorond FA, et al. Admission heart rate variability is associated with fever development in patients with intracerebral hemorrhage. Neurocrit Care 2019;30(2):244–50.

[24] Poncette A-S, Mosch L, Spies C, Schmieding M, Schiefenhövel F, Krampe H, et al. Improvements in patient monitoring in the intensive care unit: survey study. J Med Internet Res 2020;22(6):e19091.

[25] Belle A, Ansari S, Spadafore M, Convertino VA, Ward KR, Derksen H, et al. A signal processing approach for detection of hemodynamic instability before decompensation. PLOS ONE 2016;11(2).

[26] Barlow RE, Bartholomew D, Bremner JM, Brunk HD. Statistical inference under order restrictions: the theory and application of isotonic regression, Wiley series in probability and mathematical statistics, vol. 8. Wiley; 1972.

[27] Davies PL, Kovac A. Local extremes, runs, strings and multiresolution. Ann Stat 2001:1–48.

[28] Bayram I, Selesnick IW. On the dual-tree complex wavelet packet and *M*-band transforms. IEEE Trans Signal Process 2008;56(6):2298–310.

[29] Laurin A. BP_annotate. 2019 [accessed 01.01.19], https://www.mathworks.com/matlabcentral/fileexchange/60172-bp_annotate.

[30] Luo Y. The severity of stages estimation during hemorrhage using error correcting output codes method, VCU digital archives. Richmond, Virginia: Virginia Commonwealth University; 2012.

[31] Tucker LR. Implications of factor analysis of three-way matrices for measurement of change. Probl Meas Change 1963;15:122–37.

[32] Tucker LR, et al. The extension of factor analysis to three-dimensional matrices. Contrib Math Psychol 1964;110119.

[33] De Lathauwer L, De Moor B, Vandewalle J. A multilinear singular value decomposition. SIAM J Matrix Anal Appl 2000;21(4):1253–78.

[34] Kolda TG, Bader BW. Tensor decompositions and applications. SIAM Rev 2009;51 (3):455–500.

[35] Hillar CJ, Lim L-H. Most tensor problems are NP-hard. J ACM (JACM) 2013;60(6): 1–39.

[36] Pearson K. LIII. on lines and planes of closest fit to systems of points in space. Lond Edinb Dublin Philos Mag J Sci 1901;2(11):559–72.

[37] Hotelling H. Relations between two sets of variates. Biometrika 1936;28(3/4): 321–77.

[38] Ding C, Peng H. Minimum redundancy feature selection from microarray gene expression data. J Bioinform Comput Biol 2005;3(02):185–205.

[39] Peng H, Long F, Ding C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 2005;27(8):1226–38.

[40] Breiman L. Random forests. Mach Learn 2001;45(1):5–32.

[41] Maron ME. Automatic indexing: an experimental inquiry. J ACM (JACM) 1961;8 (3):404–17.

[42] Cortes C, Vapnik V. Support-vector networks. Mach Learn 1995;20(3):273–97.

[43] Freund Y, Schapire R, Abe N. A short introduction to boosting. J-Jpn Soc Artif Intell 1999;14(771-780):1612.

[44] Sabeti E, Gryak J, Derksen H, Biwer C, Ansari S, Isenstein H, et al. Learning using concave and convex kernels: applications in predicting quality of sleep and level of fatigue in fibromyalgia. Entropy 2019;21(5):442.

[45] Fan R-E, Chen P-H, Lin C-J. Working set selection using second order information for training support vector machines. J Mach Learn Res 2005;6(December): 1889–918.

[46] Friedman JH. Greedy function approximation: a gradient boosting machine. Ann Stat 2001:1189–232.

[47] Hall MA. Correlation-based feature selection for machine learning. 1999.