

Learning Predictive and Interpretable Timeseries Summaries from ICU Data

Nari Johnson, MSc, Sonali Parbhoo, PhD, Andrew S Ross, PhD, Finale Doshi-Velez, PhD
School of Engineering and Applied Sciences, Harvard University, Cambridge, MA

Abstract

Machine learning models that utilize patient data across time (rather than just the most recent measurements) have increased performance for many risk stratification tasks in the intensive care unit. However, many of these models and their learned representations are complex and therefore difficult for clinicians to interpret, creating challenges for validation. Our work proposes a new procedure to learn summaries of clinical timeseries that are both predictive and easily understood by humans. Specifically, our summaries consist of simple and intuitive functions of clinical data (e.g. “falling mean arterial pressure”). Our learned summaries outperform traditional interpretable model classes and achieve performance comparable to state-of-the-art deep learning models on an in-hospital mortality classification task.

1 Introduction

Accurate predictions of patient risk in critical care units can aid clinicians in making more effective decisions. Specifically, early identification of patients at high risk for in-hospital mortality is critical to assess patient disease acuity and inform life-saving interventions [1,2]. To predict in-hospital mortality risk, researchers have developed algorithms ranging from simple score-cards [1,3] to statistical machine learning (ML) models. Recent advances in ML have led to the development of models with vast improvements in predictive accuracy for patient in-hospital mortality risk [4–7].

Despite these improvements, however, ML models are still prone to critical errors, often failing to generalize across different care settings or institutions [8]. An emerging line of research in interpretability, defined by [9] as “the ability to explain or to present in understandable terms to a human,” provides an alternative way to ensure systems preserve properties such as safety or nondiscrimination. If a model is interpretable to stakeholders, then clinical experts can inspect the model and verify that its reasoning is sound. This ability to audit and validate is especially important when the models are used to inform critical decisions affecting patient health.

In this work, we present a novel ML method to learn clinical timeseries summaries that are interpretable and predictive. We introduce functions to compute summaries that align with simple and intuitive concepts, such as whether the timeseries is decreasing or spikes above a critical threshold. In contrast to prior work, our method learns how much of the timeseries should be used to calculate these summaries, discarding earlier timesteps that may be irrelevant for a specific prediction task. Importantly, we introduce relaxations of our summary definitions to enable differentiable optimization, allowing summary parameters to be learned jointly with those of a downstream model. We show that with our method, we can achieve accuracies comparable with state-of-the-art baselines without sacrificing interpretability.

2 Related Work

Prior work on explaining clinical timeseries models fall into two categories: learning simple models that are inherently interpretable, and generating explanations of complex black box models. We summarize a few key examples below.

Interpreting Deep Models. One popular strategy for explaining ML models is learning a second post-hoc explanation model to explain the first black box model [10]. Many post-hoc explanation techniques for clinical timeseries models train explanation models that quantify the relative importance of each clinical variable [11, 12]. However, several works argue against the use of post-hoc explanation techniques, as explanation models are not always faithful or representative of the true underlying black boxes [13, 14]. Our method avoids these problems by design, instead explicitly optimizing for interpretability so that a second explanation model is not needed.

Another line of research proposes attention mechanism models specifically designed for timeseries. [15, 16] present neural attention architectures for clinical timeseries and argue that attention scores measure feature importance. However, attention methods are often highly complex and nonlinear. Furthermore, [17] shows attention scores do not always reflect true importance. Instead of approximating importance, our study uses linear models over richer fea-

tures, where importance does not need to be approximated but can be directly read off model coefficients.

Expert Systems and Expert Features. Our work extends a long tradition of clinical experts hand-crafting features to create interpretable clinical decision-support algorithms. Two expert systems widely used in ICUs are SAPS-II [1] and APACHE [3], which use simple score-card algorithms to evaluate patient acuity. These systems use input features such as the patient’s average or worst lab or vital values over time to compute mortality risk. While SAPS-II and APACHE are simple and simulable to clinicians, their predictiveness is limited, as they cannot capture how labs or vitals change across a patient’s stay.

A similar line of research proposes manual construction of expert features from clinical data, which are then used as input to ML models [18, 19]. One limitation of this approach is that expert feature derivation is expensive and requires clinical expertise. Rather than relying on expert knowledge to identify which summary features will be the most predictive for a given task, our work instead uses optimization with a sparsity constraint to automatically learn which summary features are the most predictive.

Summarizing Clinical Timeseries. A number of works have proposed a wide range of summary statistics for patient timeseries data. Many works such as [4, 20] train clinical models using the minimum value, maximum value, first measured value, or skew of clinical timeseries data. [21] proposes a more comprehensive set of 14 summary statistics to characterize the central tendency, dispersion tendency, and distribution shape of clinical timeseries data. Our work extends this research, and is the first to our knowledge to use the slope of the timeseries or proportion of time above or below critical thresholds. Our work is also novel in that we explicitly model and optimize for the duration over which we compute each summary feature.

3 Cohort and Problem Set-Up

Our goal is to learn summaries of patient timeseries data that are both human-interpretable and predictive for a downstream classification task. Our approach will define how to calculate these human-interpretable summaries, and describe how both summary and classification model parameters are learned through optimization. In what follows, we detail each of these processes.

Prediction Task. Our work examines early prediction of in-hospital mortality. We use the patient’s first 24 hours of data to predict if they would later expire over the course of the remainder of their admission. Patients who expired in the first 24 hours of their stay were excluded from our cohort.

Cohort Selection. We use data from the MIMIC-III critical care database [22], which contains deidentified health data from patients in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. All data was extracted from MIMIC-III PhysioNet version 1.4, which contains 30,232 patients. We exclude patients under 18 years of age and patients whose weight is not measured. We include data from each patient’s first hospitalization only, and only patients with stays between 24-72 hours [4]. After applying these criteria, our final cohort contained 11,035 patients, 15.23% of whom died in-hospital. Cohort characteristics and demographics are summarized in Table 1.

| Cohort | Age | % Female | % Urgent | % Emergency | % Elective | % MICU | % SICU | % CCU | % CSRU |
|--------|------|----------|----------|-------------|------------|--------|--------|-------|--------|
| All | 64.7 | 43.8 | 1.12 | 84.46 | 14.43 | 42 | 18 | 12 | 16 |
| + | 70.9 | 46.5 | 0.77 | 96.25 | 2.97 | 53 | 18 | 13 | 4 |
| - | 64.1 | 43.6 | 1.14 | 83.22 | 15.64 | 41 | 18 | 12 | 17 |

Table 1: Mean statistics for the population cohort, and for cohorts of positive versus negative patients for in-hospital mortality. Abbreviations: MICU, medical care unit; SICU, surgical care unit; CCU, cardiac care unit; CSRU, cardiac-surgery recovery unit.

3.1 Extracting Inputs and Outputs

For each patient n in our N patient cohort, we extracted static observations and physiological data including labs and vital signs sampled hourly. All clinical variables were separately normalized to have zero mean and unit variance. Figure 1 shows how features are extracted for patients that are positive versus negative for in-hospital mortality.

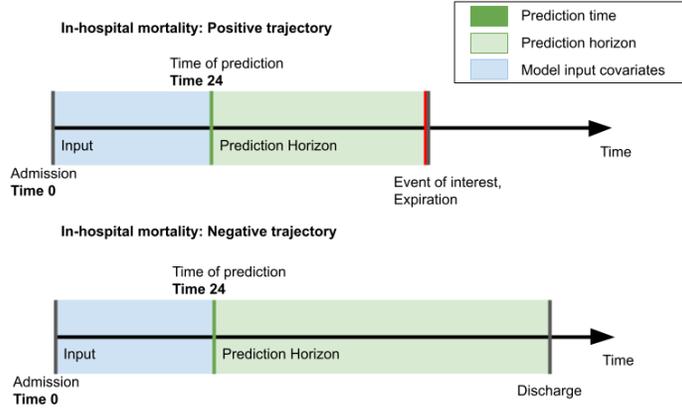


Figure 1: Example positive and negative time-series to illustrate feature extraction. The two trajectories have input data extracted from time 0 to time of prediction $T = 24$. Figure inspired by Sherman et al [23].

Static observations S . Matrix $S : (N \times 8)$ contains 8 demographic variables for each patient n : their age at admission, gender, and other information about their ICU stay (their first ICU service type, and whether their admission was urgent, emergency, or elective).

Per-timestep clinical observations X . The clinical variable tensor $X : (N \times D \times T)$ contains $D = 28$ measurements of clinical variables for each of the N patients at time t , discretized by hour. These 28 measurements consist of vital signs and labs: diastolic blood pressure, fio2, GCS score, heartrate, mean arterial blood pressure, systolic blood pressure, SRR, oxygen saturation, body temperature, urine output, blood urea nitrogen, magnesium, platelets, sodium, ALT and AST, hematocrit, po2, white blood cell count, bicarbonate, creatinine, lactate, pco2, glucose, INR, hemaglobin, and bilirubin. Missing values at timestep t were imputed using either the most recent measurement of the variable, or the population median if the variable had not yet been measured during the patient’s stay. We use the patient’s first $T = 24$ hours of data to predict in-hospital mortality. We use subscripts to index into the tensor: for example, X_t indicates the $(N \times D)$ matrix of measurements taken at time t .

Per-timestep measurement indicators M . The measurement indicator tensor $M : (N \times D \times T)$ contains indicator elements $M_{n,d,t}$ which are 1 if their corresponding clinical variable in $X_{n,d,t}$ was measured at time t , 0 otherwise.

Outcome labels y . Label vector y contains indicators y_n which are 1 if patient n expired in-hospital, 0 otherwise.

4 Methods

Given a cohort of N training examples $\{X, M, S, y\}$, we propose a novel procedure for learning predictive human-interpretable timeseries summaries. Concretely, we first compute summaries H from clinical timeseries data (X, M) . We then use the summaries H in addition to static data S and clinical variables X as input to a Logistic Regression model to predict labels y . This process of using human-interpretable summaries for prediction is shown in Figure 2.

In Section 4.1, we motivate and introduce our novel summary features. In Section 4.2, we give continuous relaxations of summary feature definitions to enable efficient inference of summary parameters. In Section 4.3, we discuss how predictive summary features can be jointly learned with downstream classification model parameters.

4.1 Model: Defining human-interpretable summaries

Measures of central tendency and dispersion have commonly been used to summarize timeseries (see survey in [21]). Our key modelling innovations include adding additional features that correspond to how clinicians themselves describe timeseries. Our novel contributions include explicitly modelling the overall trend of a lab/vital and the number of hours that a lab/vital dips above or below a threshold, as well as allowing different features to be computed over different periods of time (e.g. the most recent 6 hours vs. the most recent 24 hours).

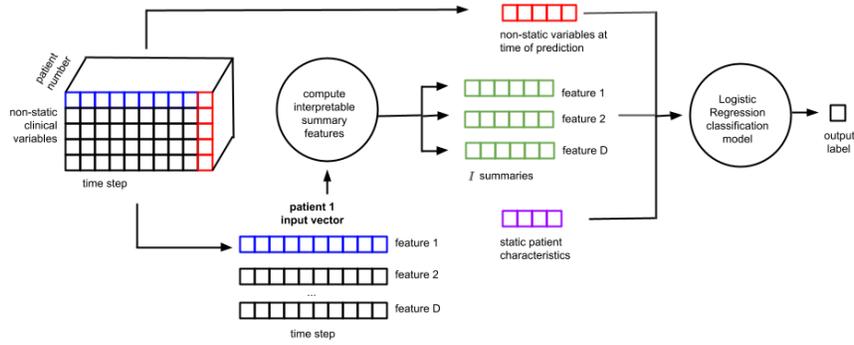


Figure 2: Illustrates summary extraction for prediction from timeseries data. First, non-static clinical variables $\{\mathbf{X}, \mathbf{M}\}$ are used to compute interpretable summary features \mathbf{H} . Then, summary features \mathbf{H} , static features \mathbf{S} , and the non-static variables $\{\mathbf{X}, \mathbf{M}\}$ at the time of prediction are given as input to a Logistic Regression model g , which predicts output labels $\hat{\mathbf{y}}$. Figure inspired by Ghassemi, Szolovits et al [24].

The $I = 13$ summary statistics used in this study are listed in Table 2. Each of the summary statistics takes into account measurement indicators \mathbf{M} so that clinical variable summaries are computed only using timesteps where the variable is measured. Each summary statistic is applied to each of the D clinical variables to create the summary feature tensor $\mathbf{H} : (N \times D \times I)$.

Below, we expand on the parameterization of our summary features. Next, we describe how we enable efficient, automated search over summary feature parameters. Importantly, our approach automates many processes associated with summary design, enabling optimization over summary parameters.

Incorporating Duration. Many prior works in clinical timeseries modelling do not use all timesteps for the patient, but instead only the most recent available data, such as the six or twelve hours before the time of prediction. In contrast to prior works, we explicitly model how much of each timeseries should be used to compute each of the I summaries in \mathbf{H} . For example, we may wish to exclude earlier measurements of a particular clinical variable if only the variable’s recent measurements before time of prediction T are significant for a prediction task. Specifically, for each variable d and for each summary function i , we define a duration time $C_{d,i}$. Only the variable’s timeseries data that occurred in the immediately previous $C_{d,i}$ hours before the time of prediction is used to calculate summary i . We organize all the duration time parameters $C_{d,i}$ into a $(D \times I)$ matrix \mathbf{C} .

To exclude data that occurs before time $(T - C_{d,i})$ when computing summary features, we multiply each of the original timeseries variables by indicator variables for whether the measurements occurred within $C_{d,i}$ hours before time of prediction T . For example, a mean summary statistic would be computed using indicator variables $\mathbb{1}(\cdot)$ as:

$$\mathbf{H}_{i=mean} : (N \times D) = \left(\sum_{t=1}^T \mathbb{1}(t > T - C_{i=mean}) \odot \mathbf{X}_t \odot \mathbf{M}_t \right) / \left(\sum_{t=1}^T \mathbf{M}_t \odot \mathbb{1}(t > T - C_{i=mean}) \right) \quad (1)$$

where \odot is the element-wise multiplication operator and division is performed element-wise. Our objective is to learn duration parameters \mathbf{C} that maximize the predictiveness of their corresponding summary features \mathbf{H} .

Threshold Parameters. Some of the summary functions f_i in Table 2 have additional parameters such as thresholds. For example, one of the summaries is the proportion of the patient’s measured timeseries where their measured clinical variables are above some D -dimensional critical threshold parameter vector ϕ^+ for each variable:

$$\left(\sum_{t=1}^T \mathbf{M}_t \odot \mathbb{1}(\mathbf{X}_t > \phi^+) \right) / \left(\sum_{t=1}^T \mathbf{M}_t \right) \quad (2)$$

These summaries correspond to clinically-intuitive ideas, such as whether the patient been mostly well or sick. As with durations, we learn threshold parameters automatically to avoid burdening experts and to assist in prediction.

4.2 Continuous Relaxations for Efficient Inference

Learning Duration Parameters. In Equation 1, we showed how summary functions f_i that only use the most recent C hours of data can be calculated using indicator variables $\mathbb{1}(t > T - C_{d,i})$. These indicator variables, however, do not have informative gradients and are not differentiable. To enable differentiable optimization for duration time parameters C , we introduce weight parameters \mathbf{W} by relaxing the indicator random variables using the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$. Using the duration parameter matrix C , we define D -dimensional vectors $\mathbf{w}_{t,i}$ that compose weight tensor $\mathbf{W} : (T \times I \times D)$:

$$\mathbf{w}_{t,i} = \sigma((t - T + C_i)/\tau) \quad (3)$$

For each feature d and summary i , clinical observations $\mathbf{X}_{d,t}$ where $t > T - C_{d,i}$ will have corresponding weights $\mathbf{w}_{t,i,d}$ near 1. Timesteps t where $t < T - C_{d,i}$ will have corresponding weights $\mathbf{w}_{t,i,d}$ near 0. Temperature parameter τ controls the harshness of the weight matrix: small temperatures push the sigmoid function towards its edges, learning weights that are closer to exactly 0 for timesteps before $T - C_{d,i}$ and 1 for timesteps after $T - C_{d,i}$, effectively functioning as the indicator variables in Equation 1.

Weighted summary functions f_i used to derive human-interpretable summaries \mathbf{H} can be found in Table 2. Duration parameters C that determine weight tensor \mathbf{W} are included in $\beta_{\mathbf{H}}$, the set of all parameters necessary to compute the summaries.

| Description | Function |
|---|---|
| Mean of the time-series | $\left(\sum_{t=1}^T (\mathbf{w}_{t,i} \odot \mathbf{X}_t \odot \mathbf{M}_t) \right) / \left(\sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i} \right)$ |
| Variance of the time-series | $\frac{(\sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i})^2}{(\sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i})^2 - \sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_t^2} \odot \sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i} \odot (\mathbf{X}_t - \bar{\mathbf{X}})^2$ |
| Indicator if feature was ever measured | $\sigma \left(\left(\sum_{t=1}^T \mathbf{w}_{t,i} \odot \mathbf{M}_t \right) / \left(\tau \odot \sum_{t=1}^T \mathbf{w}_{t,i} \right) \right)$ |
| Mean of the indicator sequence | $\left(\sum_{t=1}^T \mathbf{w}_{t,i} \odot \mathbf{M}_t \right) / \left(\sum_{t=1}^T \mathbf{w}_{t,i} \right)$ |
| Variance of the indicator sequence | $\left(\frac{(\sum_{t=1}^T \mathbf{w}_{t,i})^2}{(\sum_{t=1}^T \mathbf{w}_t)^2 - \sum_{t=1}^T \mathbf{w}_t^2} \right) \sum_{t=1}^T \mathbf{w}_{t,i} \odot (\mathbf{M}_t - \bar{\mathbf{M}})^2$ |
| # switches from missing to measured | $\left(\sum_{t=1}^{T-1} \mathbf{w}_{t,i} \odot \mathbf{M}_{t+1} - \mathbf{M}_t \right) / \left(\sum_{t=1}^T \mathbf{w}_{t,i} \right)$ |
| First time the feature is measured | $\min t \text{ s.t. } \mathbf{M}_t = 1$ |
| Last time the feature is measured | $\max t \text{ s.t. } \mathbf{M}_t = 1$ |
| Proportion of time above threshold ϕ^+ | $\left(\sum_{t=1}^T \mathbf{w}_{t,i} \odot \mathbf{M}_t \odot \sigma \left(\frac{\mathbf{X}_t - \phi^+}{\tau} \right) \right) / \left(\sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i} \right)$ |
| Proportion of time below threshold ϕ^- | $\left(\sum_{t=1}^T \mathbf{w}_{t,i} \odot \mathbf{M}_t \odot \sigma \left(\frac{\phi^- - \mathbf{X}_t}{\tau} \right) \right) / \left(\sum_{t=1}^T \mathbf{M}_t \odot \mathbf{w}_{t,i} \right)$ |
| Slope of a L2 line | $\frac{\sum_{t=1}^T \mathbf{w}_{t,i} (t - \bar{t}_w) (\mathbf{X}_t - \bar{\mathbf{X}}_w)}{\sum_{t=1}^T \mathbf{w}_{t,i} (t - \bar{t}_w)^2}$, where $\bar{t}_w = \frac{\sum_t \mathbf{w}_{t,i} \cdot t}{\sum_t \mathbf{w}_{t,i}}$ and $\bar{\mathbf{X}}_w = \frac{\sum_t \mathbf{w}_{t,i} \odot \mathbf{X}_t}{\sum_t \mathbf{w}_{t,i}}$ |
| Standard error of the L2 line slope | $1 / \left(\sum_t \mathbf{w}_{t,i} \odot (t - \bar{t}_w)^2 \right)$ |

Table 2: Table of functions f_i used to calculate human-interpretable summaries \mathbf{H} . For each of the D clinical variables, all I of the above functions are applied to each of the N patients. Each of the I summary features i is defined with respect to D -dimensional weight vectors $\mathbf{w}_{t,i}$ defined in Section 4.2. Parameter τ is a temperature parameter for the sigmoid function. $\mathbb{1}(\cdot)$ denotes indicator variables for events inside the parentheses, \odot indicates element-wise matrix multiplication and division is done element-wise. Additionally, $\bar{\mathbf{X}} = \sum_t^T \mathbf{M}_t \odot \mathbf{X}_t / \sum_t^T \mathbf{M}_t$, and $\bar{\mathbf{M}}_t = \frac{1}{T} \sum_t^T \mathbf{M}_t$.

Learning Threshold Parameters. Our work relaxes summary definitions to enable differentiable optimization to learn summary parameters. The indicator variables used to define the proportion of hours that a patient’s timeseries is above thresholds ϕ^+ in Equation 2 are non-differentiable with respect to ϕ^+ . To enable differentiable optimization,

our work defines our threshold summary features using the sigmoid function σ with temperature parameter τ :

$$f_{threshold}(\mathbf{X}, \mathbf{M}, \mathbf{W}) = \left(\sum_{t=1}^T \mathbf{w}_{t,i=threshold} \odot \mathbf{M}_t \odot \sigma \left(\frac{\mathbf{X}_t - \phi^+}{\tau} \right) \right) / \left(\sum_{t=1}^T \mathbf{M}_t \cdot \mathbf{w}_{t,i=threshold} \right) \quad (4)$$

Threshold parameters $\{\phi^+, \phi^-\}$ are included in β_H , the set of all parameters necessary to compute the summaries.

4.3 Learning Process

Our study uses summary features H , along with static variables S and timeseries variables $\{\mathbf{X}, \mathbf{M}\}$ at prediction time $T = 24$ as input to a Logistic Regression model g with coefficients β_g . Logistic Regression model g outputs predicted probabilities $\hat{\mathbf{y}} = p(\mathbf{y} = 1 | \mathbf{X}, \mathbf{S}, \mathbf{M})$. Our objective is to learn optimal summary and model parameters $\beta = \{\beta_H, \beta_g\}$. We jointly learn parameters β by minimizing the loss function

$$\mathcal{L}(\beta; \mathbf{X}, \mathbf{M}, \mathbf{S}, \mathbf{y}) = -\frac{1}{N} \sum_{n=1}^N \omega_n (\mathbf{y}_n \cdot \log[g(\mathbf{X}_n, \mathbf{M}_n, \mathbf{S}_n, \beta)] + (1 - \mathbf{y}_n) \log[1 - g(\mathbf{X}_n, \mathbf{M}_n, \mathbf{S}_n, \beta)]) + \Omega(\beta_g) \quad (5)$$

Our loss function is the sum of the weighted binary cross-entropy loss using predictive model g and the regularization penalty $\Omega(\beta_g)$. To account for class imbalance, we reweight each training example n 's loss contribution by ω_n , the inverse of its class frequency in the training dataset.

Horseshoe Regularization on Coefficients. We explicitly optimize for sparsity in model parameters β_g via our regularization penalty in our training objective. We use a Horseshoe regularization penalty $\Omega(\beta_g)$ with shrinkage parameter 1 to encourage sparsity in the learned regression coefficients [25].

5 Experiments

We compare models learned using our summaries to other interpretable models as well as deep learning baselines. We show that our models outperform other traditional human-interpretable model classes and achieve performance comparable to deep models on the in-hospital mortality task.

Model configurations. Our baseline models are: Ridge Logistic Regression models that take as input only the patient timeseries measured at the time of prediction T , and Ridge Logistic Regression and LSTM models that take as input all of the patient timeseries. We used an LSTM as our deep baseline architecture as prior works document their superior performance at mortality prediction from clinical timeseries data [26].

Our LSTM models were trained on the sequential timeseries data $\{\mathbf{X}, \mathbf{M}\}$ with a step size of 1 hour. The LSTM hidden states at each timestep were then used to predict both the next timestep of the patient timeseries \mathbf{X} , and to predict outcome labels \mathbf{y} . All T of the output hidden states \mathbf{X}_t were input to a fully-connected layer, which output predictions of the next timestep \mathbf{X}_{t+1} . The last output hidden state at time T was also input with static features \mathbf{S} to a fully-connected layer to predict outcome labels \mathbf{y} . The LSTM models were trained to minimize both the mean-squared error of the next state prediction, and the binary cross entropy loss of the classification prediction. ReLU activation functions were applied to both fully-connected layers.

Training Details. For training and testing, we split the cohort of N patients into train and test sets, where all data associated with each patient is either in train or test. All performance metrics are averaged across five train-test splits. Ridge baseline models were implemented using RidgeCV from `scikit-learn` [27]. LSTMs as well as our summary-based Logistic Regression models were implemented with PyTorch, and trained with the Adam optimizer [28] at a batch size of 256. We trained all of our Logistic Regression models for 30,000 epochs and LSTM models for 10,000 epochs using early stopping. All hyperparameters (including the LSTM hidden state and layer dimensions, optimizer learning rate, and regularization parameters) were selected via random hyperparameter search [29]. All temperature parameters τ were set to 0.1.

Our final learned LSTM models have hidden state dimension 32, 1024 nodes in the layer to predict the next timestep, and 64 nodes in the layer to predict labels \mathbf{y} . They are trained using a learning rate of $1e - 05$. Our final learned models with summaries use $\alpha = 1e - 05$, Horseshoe shrinkage parameter 1.0, and learning rate $1e - 05$.

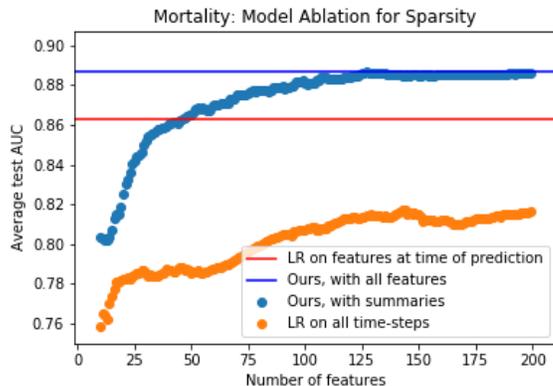


Figure 3: Prediction quality (mean test AUC) vs. model complexity (number of non-zero features) for baseline Ridge Regression trained on all timesteps of the patient timeseries, versus our models. The dark blue horizontal line shows our model’s average test set AUC using all 401 derived features, and the orange horizontal line shows the average test set AUC of the Logistic Regression model trained only on the 65 features extracted at the time of prediction.

6 Results

Our learned models achieve performance comparable to state-of-the-art baselines. Table 3 compares the performance of our learned models with linear and deep baselines for the in-hospital mortality prediction task. Applying a linear model to the learned summary features H consistently improves AUCs in comparison to using a linear model on clinical timeseries and static data alone. Our models achieve an AUC performance comparable to state-of-the-art LSTM models with test AUCs of 0.9000 ± 0.0223 . Notably, our models that allow differentiable optimization to learn duration times outperform models that compute all summary features using the entire duration of the patient timeseries. This implies that there is predictive value in explicitly modelling how much of each variable’s clinical timeseries should be considered for a specific prediction task.

| Model | Train set AUC | Test set AUC |
|--|---------------------------------------|---------------------------------------|
| LR trained on $\{S, M\}$ and X at time T only | 0.8653 ± 0.0013 | 0.8626 ± 0.0079 |
| LR trained on $\{S, M, X\}$ | 0.8931 ± 0.0015 | 0.8668 ± 0.0122 |
| LSTM trained on $\{S, M, X\}$ | 0.9101 ± 0.0018 | 0.9000 ± 0.0223 |
| Our model, trained on $\{S, M, X\}$, non-differentiable durations C | 0.9065 ± 0.0019 | 0.8818 ± 0.0063 |
| Our model, trained on $\{S, M, X\}$, differentiable durations C | 0.9074 ± 0.0016 | 0.8867 ± 0.0061 |

Table 3: Performance of learned models on the in-hospital mortality prediction task. AUCs are averaged over five train-test splits with their standard error. Abbreviations: LR, Logistic Regression; LSTM, long short-term memory. C refers to the duration parameters defined in Section 4.2.

Our learned models use fewer features to achieve higher accuracy in comparison to other interpretable baselines. To evaluate the sparsity of each model, we performed a set of ablation experiments where we zeroed all but the N coefficients with the largest magnitudes for each of the learned Logistic Regression models. In Figure 3, we show the average test set AUC for Logistic Regression baseline versus our models using only N coefficients. Our models consistently outperform baseline models when all but N coefficients are zeroed, suggesting that our models learn a smaller and more predictive set of important features.

Our learned models are interpretable. Table 4 shows 15 key summary features that consistently have the largest learned Logistic Regression coefficients across train-test splits. The corresponding coefficients for each feature can be interpreted as a measure of the feature’s contribution to the final classification label. For example, because the mean of the patient’s GCS has a large negative coefficient, this means that patients with higher mean GCS scores will be assigned lower predicted probabilities for in-hospital mortality. Therefore our models are *decomposable* [30], as each

of the model’s features and coefficients has an intuitive clinical explanation.

Our learned summary features are clinically sensible. The vast majority of the key summary features learned by our models shown in Table 4 are supported by studies in medical literature. For instance, it is widely accepted that patients who are older tend to have lower chances of survival in ICU settings [31, 32]. Similarly, patients with lower GCS scores of below 6 tend to have severe injuries and higher chances of mortality [33]. Notably a lower GCS score in the later hours of a patient’s hospitalisation significantly reduces a patient’s chances of survival [34, 35]. Finally, the normal range of features such as the blood oxygen saturation (SPO₂) is between 95% and 100%. An SPO₂ consistently below 90% indicates hypoxaemia or potential respiratory distress. These patients have to be mechanically ventilated in ICU and frequently have lower chances of survival [36, 37].

| Feature | Aggregation | Time | Coefficient |
|------------------|---------------------------|---------------|-------------|
| Age | static value | - | 14.9 |
| BUN | value at | hour 24 | 6.2 |
| GCS | mean over | hours 5 - 24 | - 5.59 |
| HR | value at | hour 24 | 4.69 |
| FiO ₂ | value at | hour 24 | 4.31 |
| Hct | times measured over | hours 2 - 24 | - 3.81 |
| HR, | mean over | hours 2 - 24 | 3.78 |
| GCS | value at | hour 24 | - 3.62 |
| GCS | hours below 6.08 | hours 7 - 24 | 3.37 |
| Creatinine | hours below 0.35 mg/dL | hours 5 - 24 | 2.76 |
| FiO ₂ | hours above 62.96% | hours 16 - 24 | 2.59 |
| SpontaneousRR | mean over | hours 2 - 24 | 2.29 |
| GCS | times measured over | hours 5 - 24 | 2.23 |
| Sodium | hours below 131.57 mEq/L | hours 1 - 24 | 2.16 |
| WBC | hours below 0.78 cells/mL | hours 6 - 24 | 2.10 |
| SPO ₂ | hours below 92.36% | hours 10 - 24 | 2.04 |

Table 4: Key summary features, sorted from largest to smallest coefficient magnitudes, from learned models.

Initialization sensitivity. In general, we observed that our optimization procedure is stable, learning the same 15 key summary features across different stochastic parameter initializations and train-test splits. However, there are cases where we observed that the learned duration parameters C varied depending on their initialization. As such, we recommend that practitioners incorporate prior knowledge about the clinical prediction task when initializing the duration time parameters. For example, if examining the entire duration of the patient’s timeseries is necessary for a prediction task, then the duration parameters should be initialized to include the entire timeseries by default.

7 Discussion & Conclusion

In this work, we defined functions to compute interpretable, parameterizable summaries of clinical timeseries, and developed relaxations so that our summary parameters could be jointly learned with a downstream predictive model. In our experiments, we used Logistic Regression to make predictions because its coefficients are easily decomposable [30]. However, because our learned summaries are inherently interpretable, any other interpretable architecture could be used instead. Our methodology is generalizable and enables the efficient learning of intuitive and predictive timeseries summaries without placing any assumptions on the downstream model architecture.

Future work. Our study poses many interesting directions for future work. One avenue would be to conduct a user study to validate the human-interpretability and decomposability of our proposed summary features. Another would be to evaluate whether the summary features learned for particular critical care prediction tasks remain predictive for a wider set of critical care prediction tasks. Finally, we could also develop additional summary statistic functions, or expand our framework to consider sharing duration parameters across features or across summaries to better model dependencies between clinical labs and vitals—as many physiological events are characterized by several simultaneous

changes to multiple labs and vitals [38].

Conclusion. In this paper, we propose a new method to learn interpretable and predictive summary features from clinical timeseries data. In addition to introducing novel summary statistics including slope and threshold features, our work differs from prior work by learning the duration of timeseries data that should be used to compute each summary. We demonstrate that our learned timeseries summaries achieve performance quality comparable to state-of-the-art deep models when trained to predict early patient mortality risk on real patient data. We also qualitatively validate our models to confirm their interpretability and sensibility. Our work is an important step towards optimizing for representations of clinical timeseries data that are both highly predictive and interpretable.

Acknowledgements: NJ and FDV acknowledge support from NIH R01 MH123804-01A1. SP acknowledges support from the Miami Foundation and SNSF P2BSP2-184359.

References

- [1] JR Le Gall, S Lemeshow, and F Saulnier. A new simplified acute physiology score (saps ii) based on a european/north american multicenter study. *JAMA*, 1993.
- [2] G Escobar, V Liu, A Schuler, B Lawson, J Greene, and P Kipnis. Automated identification of adults at risk for in-hospital clinical deterioration. *N Engl J Med*, 2020.
- [3] WA Knaus, DP Wagner, EA Draper, JE Zimmerman, M Bergner, and PG Bastos et al. The apache iii prognostic system. risk prediction of hospital mortality for critically ill hospitalized adults. *Chest*, 1991.
- [4] A Awad, M Bader-El-Den, J McNicholas, and J Briggs. Early hospital mortality prediction of intensive care unit patients using an ensemble learning approach. *IJMI*, 2017.
- [5] M Ghassemi et al. A multivariate timeseries modeling approach to severity of illness assessment and forecasting in icu with sparse, heterogeneous clinical data. *AAAI*, 2015.
- [6] A Rajkomar et al. Scalable and accurate deep learning for electronic health records. *Digital Medicine*, 2018.
- [7] S Purushothama et al. Benchmarking deep learning models on large healthcare datasets. *J.Biomed.Inform.*, 2018.
- [8] J Futoma et al. The myth of generalisability in clinical research and machine learning in health care. *Lancet DH*, 2020.
- [9] Doshi-Velez and Kim. Towards a rigorous science of interpretable machine learning. *arXiv:1702.08608*, 2017.
- [10] MT Ribeiro, S Singh, and C Guestrin. “why should i trust you?” explaining the predictions of any classifier. In *ACM SIGKDD*, 2016.
- [11] S Tonekaboni et al. What went wrong and when? instance-wise feature importance for time-series black-box models. In *NeurIPS*, 2020.
- [12] S Lundberg et al. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, 2018.
- [13] C Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *arXiv:1811.10154v3*, 2019.
- [14] D Shack, S Hilgard, E Jia, S Singh, and H Lakkaraju. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. *AIES*, 2020.
- [15] E Choi, M Taha Bahadori, J Kulas, A Schuetz, W Stewart, and J Sun. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *arXiv:1608.05745v4*, 2017.
- [16] Y Sha and M Wang. Interpretable predictions of clinical outcomes with an attention-based recurrent neural network. *ACM BCB*, 2017.

- [17] S Serrano and N Smith. Is attention interpretable? *arXiv:1906.03731v1*, 2019.
- [18] J Sun, J Hu, D Luo, M Markatou, F Wang, and S Edabollahi. Combining knowledge and data driven insights for identifying risk factors using electronic health records. *AMIA*, 2012.
- [19] K Roe et al. Feature engineering with clinical expert knowledge: A case study assessment of machine learning model complexity and performance. *PLoS ONE*, 2020.
- [20] H Harutyunyan et al. Multitask learning and benchmarking with clinical time series data. *arXiv:1703.07771v3*, 2019.
- [21] C Guo, M Lu, and J Chen. An evaluation of time series summary statistics as features for clinical prediction tasks. *BMC Medical Informatics Decis. Mak.*, 20(1):48, 2020.
- [22] AEW Johnson and TJ Pollard et al. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 2016.
- [23] E Sherman et al. Leveraging clinical time-series data for prediction: A cautionary tale. *AMIA*, 2018.
- [24] M Ghassemi, M Wu, MC Hughes, P Szolovits, and F Doshi-Velez. Predicting intervention onset in the ICU with switching state space models. *AMIA*, 2017.
- [25] Hrayr Harutyunyan. The horseshoe-like regularization for feature subset selection. *Nature*, 2019.
- [26] H Harutyunyan et al. Multitask learning and benchmarking with clinical time series data. *Scientific Data*, 6(1):96, 2019.
- [27] F Pedregosa et al. Scikit-learn: Machine learning in Python. *JMLR*, 2011.
- [28] D Kingma and J Ba. Adam: A method for stochastic optimization. *arXiv:1412.6980*, 2014.
- [29] J Bergstra and Y Bengio. Random search for hyper-parameter optimization. *JMLR*, 2012.
- [30] Z Lipton. The mythos of model interpretability. *arXiv:1606.03490v3*, 2017.
- [31] AB Nielsen et al. Survival prediction in intensive-care units based on aggregation of long-term disease history and acute physiology: a retrospective study of the Danish national patient registry and electronic patient records. *The Lancet Digital Health*, 2019.
- [32] L Fuchs et al. ICU admission characteristics and mortality rates among elderly and very elderly patients. *Intensive care medicine*, 38(10):1654–1661, 2012.
- [33] PG Bastos et al. Glasgow coma scale score in the evaluation of outcome in the intensive care unit: findings from the acute physiology and chronic health evaluation III study. *Critical care medicine*, 21(10):1459–1465, 1993.
- [34] RL Sacco et al. Nontraumatic Coma: Glasgow Coma Score and Coma Etiology as Predictors of 2-Week Outcome. *Archives of Neurology*, 1990.
- [35] C Settervall et al. In-hospital mortality and the Glasgow Coma Scale in the first 72 hours after traumatic brain injury. *Revista latino-americana de enfermagem*, 2011.
- [36] M Lazzerini et al. Hypoxaemia as a mortality risk factor in acute lower respiratory infections in children in low and middle-income countries: systematic review and meta-analysis. *PLoS One*, 10(9):e0136166, 2015.
- [37] ML Vold, U Aasebø, T Wilsgaard, and H Melbye. Low oxygen saturation and mortality in an adult cohort: the Tromsø study. *BMC pulmonary medicine*, 15(1):1–12, 2015.
- [38] R Hotchkiss et al. Sepsis and septic shock. *NRDP*, 2016, 2016.