TWO-STEP INTERPRETABLE MODELING OF INTENSIVE CARE ACQUIRED INFECTIONS

G. Lancia,
Mathematical Institute, Utrecht University, The Netherlands

M. Varkila,
Julius Center for Health Sciences and Primary Care, University Medical Center Utrecht, The Netherlands,
Department of Intensive Care Medicine, University Medical Center Utrecht, The Netherlands

O. Cremer,
Julius Center for Health Sciences and Primary Care, University Medical Center Utrecht, The Netherlands,
Department of Intensive Care Medicine, University Medical Center Utrecht, The Netherlands

C. Spitoni,
Mathematical Institute, Utrecht University, The Netherlands
C.Spitoni@uu.nl

ABSTRACT

We present a novel methodology for integrating *high resolution* longitudinal data with the dynamic prediction capabilities of survival models. The aim is two-fold: to improve the predictive power while maintaining interpretability of the models. To go beyond the *black box* paradigm of artificial neural networks, we propose a parsimonious and robust semi-parametric approach (i.e., a landmarking competing risks model) that combines routinely collected *low-resolution* data with predictive features extracted from a convolutional neural network, that was trained on *high resolution* time-dependent information. We then use *saliency maps* to analyze and explain the extra predictive power of this model. To illustrate our methodology, we focus on healthcare-associated infections in patients admitted to an intensive care unit.

***K*eywords** Landmarking Approach, Convolutional Neural Networks, Dynamic Prediction, ICU Acquired Infections, Saliency Maps.

1 Introduction

Although Artificial Neural Networks (ANNs) are very accurate predicting tools if compared to more conventional survival models (Topol, 2019; Zeng et al., 2022; Ivanov et al., 2022), they are often seen as *black boxes*. ANN models are indeed very difficult to interpret and it is challenging to identify which predictors are the most relevant (May et al., 2011). In contrast, semi-parametric hazard based survival models (Andersen et al. (1993)) are examples of interpretable models, whose hazards can measure (directly or indirectly) the effect of each covariate on the outcome of interest.

In order to properly model the temporal evolution of the survival process, including longitudinal information (e.g., biomarkers, health status, clinical measurements) as time-dependent covariates is often informative. These covariates are usually *internal* and they require extra modeling to predict survival functions accurately (Cortese and Andersen, 2010). The use of Joint Modeling (JM), that attempts to jointly model the longitudinal covariates and the event time, might be then a natural choice (Proust-Lima and Taylor, 2009; Rizopoulos, 2011, 2012). Despite JMs efficiently estimate the underlying parameters when the model is correctly specified, they are sensitive to misspecification of the longitudinal trajectory (Ferrer et al., 2019) and they are complex to estimate.

For these reasons, we consider a Landmarking (LM) approach for the dynamic prediction of the outcome of interest (e.g., intensive care unit acquired infections). LM is indeed a pragmatic approach that avoids specifying a model for the longitudinal covariates and it is robust under misspecification of the longitudinal processes (Van Houwelingen, 2007; van Houwelingen and Putter, 2011). The main idea behind LM is to select a point in time s known as a landmark. By selecting subjects at risk at s (i.e., left-truncation at time s) and by imposing administrative right-censoring at time $s+w$ (*horizon time*), a landmark dataset is then constructed. Thus, for a time-dependent covariate $Z(t)$, only the value $Z(s)$ at s is considered, so that the resulting LM dataset can be analyzed by using standard methods: $Z(s)$ is indeed treated as a time constant covariate. In case of competing events, the LM approach can be generalized to Competing Risks model (LM-CR), see Nicolaie et al. (2013).

The novelty of the manuscript is the inclusion in the LM-CR model of time-dependent information coming from *high-resolution* Electronic Health Record (EHR) data: vital signals recorded in the Intensive Care Unit (ICU) monitors and sampled every minute (i.e., heart rate, mean arterial blood pressure, pulse pressure, arterial oxygen saturation, and respiratory rate). A type of deep neural network, a Convolutional Neural Network (CNN), that looks for predicting patters present in the signals prior the landmark time s, is used as features' extractor to be included in the main LM-CR model. We hypothesize indeed that these patterns represent additional information, not contained in the *lower-resolution* covariates.

Although the LM-CR is in itself an interpretable model, we would like to interpret the additional predicting power of the CNN score in terms of medical conditions of the patients. Hence, we study the pattern recognition performed by the CNN, and make it interpretable via a Saliency Map Order Equivalent (SMOE) scale (Mundhenk et al., 2019), an algorithm that describes the statistics of the activated feature maps of the hidden layers of the network. By the SMOE scale we can visualize the regions of the input data with the highest *saliency* for the prediction. Hence, we extract subsets of the signal with the highest cumulative saliency, in order to perform a data-driven clustering of patients who are more likely to experience the outcome in the fore-coming prediction window. This approach represents a proof of concept for future applications of our method.

In order to illustrate the methodology, we focus on healthcare-associated infections in patients admitted to an ICU, where they are a major cause of morbidity and mortality (Vincent et al., 2009; Maki et al., 2008). Therefore, early identification of infectious events could help physicians in the prevention and management of infectious complications in the ICU (Dantes and Epstein, 2018). Moreover, the dynamic prediction of nosocomial infections is a modeling challenging task. In fact, the establishment of the presence of infection is not straightforward, and the exact time of infection onset cannot be directly observed. Hence, a method that can predict an approaching infection, might give to the partitioners valuable lead time to intervene.

The structure of the paper is the following. In Section 2 we describe the data and we define the outcome we want to predict; in Section 3 we introduce the two-step modeling approach; in Section 4 we explain the design of the CNN, its training and the *risk score*'s extraction. In Section 5 we define and fit the LM-CR model with the inclusion of the *risk score* extracted by the CNN. Finally, in Section 6 we perform a data-driven clustering based on the SMOE scale analysis of the EHR instances. The *Supplementary material* file contains further information about the data, the selection of the design of the CNN, and a more detailed explanation of the SMOE scale used in the paper.

2 The data

We analysed data from the Molecular Diagnosis and Risk Stratification of Sepsis (MARS)-cohort (Klouwenberg et al., 2013). We selected patients >18 years of age having a length-of-stay >48 hours, who had been admitted to the ICU of one of the participating study centres between 2011 and 2018. In addition, we also used high-resolution data streams from vital signs monitors which had been recorded in the hospital information system at a 1-minute resolution.

As the outcome parameter for our primary modeling attempt we used the onset of a first occurrence of a suspected ICU-AI within a 24-hour time-window from the moment of prediction. Time of infection onset was determined by either the start of new empirical antimicrobial treatment or the sampling of blood for culture (subsequently also followed by antibiotic therapy), whichever occurred first. The dataset thus consisted of 5075 ICU admissions in which 871 first cases of suspected Intensive Care Unit Acquired Infections (ICU-AIs) occurred. Importantly, the incidence of ICU-AI remained relatively constant across ICU stay at a mean rate of 0.04 (SE 0.01) events per day during the first 10 days in ICU. Median time of onset was 5.25 (IQR 3.80-9.45) days following admission.

We selected candidate predictors among several variables based on literature review, a priori consensus of clinical importance, and prevalence in the study population. These covariates include both time-fixed variables reflecting the baseline risk of infection, as well as time-dependent data representing the dynamics of the clinical evolution of patients over time, e.g., laboratory values and physiological response and organ function parameters; see Table 1 and Table 2 in Section 1 of the *Supplementary Material*.

3 Two-step modeling strategy

In order to take advantage of all longitudinal clinical data and to include observations with different temporal resolutions, we designed our model by means of a *two-step* modeling approach. In particular:

Step 1: We first use a CNN to investigate the longitudinal evolution of EHR data. In our case, the EHR data are high-frequency vital signals, recorded in the ICU monitors with a sampling frequency of 1 minute. The CNN will derive a *risk score of infection* (or more simply *risk score*), to be added to the predictors of Step 2. This extra *risk score* predictor is obtained by processing those patterns in the EHR signals that are linked to the onset of ICU-AI.

Step 2: We develop and fit a LM-CR model, including all the explanatory variables: the *baseline covariates* (e.g, sex, age, ICU admission type, and admission comorbidities); the *low-frequency predictors* (e.g., consciousness score, laboratory measurements, and bacterial colonization) and the *risk score* fitted by the CNN.

Therefore, we consider the CNN outputs as extra condensed information about the approaching of the infectious episode. Note that the CNN score is based only upon the analysis of the vital signs signal data.

4 Step 1: CNN at work

4.1 Selection of high-frequency instances

With the term *high-frequency* covariates, we refer to five vital signs signals: Heart Rate (HR), mean Arterial Blood Pressure (ABP), pulse pressure, saturation ($\mathrm{SaO2}$), and Respiratory Rate (RR). These predictors are sampled with a sampling rate equal to one minute and they are arranged like a time-series (e.g., 1440 observations for a time window of 24 hours).

We selected and extracted the *time-series instances* as follows:

1. We first remove the last 24 hours of records for all patients who died during the stay.
2. Starting from admission time τ_0^i of the patient i, we partition all physiological vital signals time-series into time windows of width $w = 24$ hours until the final time T_ℓ^i of the patient record (defined as in point 1 for the patients who died during the stay). Therefore, we obtain the set of intervals $\mathcal{P}^i$ for the patient i:

$$\mathcal{P}_i := \bigcup_{k\geq 1} \left\{ [\tau_0^i + (k-1)w, \min(\tau_0^i + kw, T_\ell^i)] \right\}$$

We define the set of time windows shifted by δ as:

$$\mathcal{P}_i^\delta := \bigcup_{k\geq 1} \left\{ [\tau_0^i + \delta + (k-1)w, \min(\tau_0^i + \delta + kw, T_\ell^i)] \right\},$$

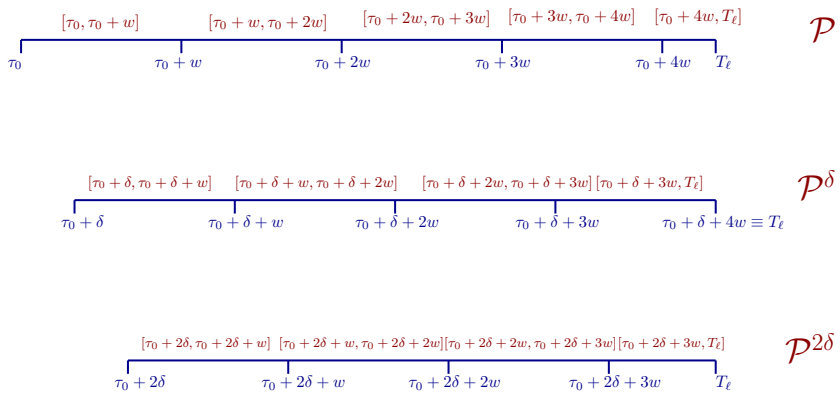


Figure 1: Example of time windows selected for one patient.

provided that $T_\ell^i \geq \tau_0^i + \delta$. Hence, the time windows selected for the patient i are the one belonging to the set $\mathcal{P}_i^{\text{total}} := \mathcal{P}_i \cup \mathcal{P}_i^{8\text{hrs}} \cup \mathcal{P}_i^{16\text{hrs}}$, see Figure 1. The collection of the time windows in $\mathcal{P}_i^{\text{total}}$ (i.e., consecutive windows of 24 hours and their translations of 8 and 16 hours), allows to chunk the longitudinal evolution of the signals coherently with the way we extracted the low-frequency time-dependent covariates of Step 2. We shall refer to the portion of the vital signs signals corresponding to an interval in $\mathcal{P}_i^{\text{total}}$ with the term *time-series instance*.

3. Per each patient i who has not acquired an infection during the stay in the ICU, we call *not-infected* instances all the instances whose time windows are in $\mathcal{P}_i^{\text{total}}$.
4. For each patient i who has acquired an infection during the stay in the ICU, we first divide the complete ICU as in point 2 ($\mathcal{P}_i^{\text{total}}$). We then label all time-windows where an ICU-AI event has occurred as an outcome event (i.e., the time-window includes the time-stamp at which the ICU-AI episode has been recorded). In addition, we also label all the time windows preceding a time-window containing the onset of an ICU-AI event as outcome events. All remaining time windows are treated as non-infected.
5. We only consider the first ICU-AI and we discard all the other recurrent episodes from the same patient. Thus, all the instances following the first infection are discarded.
6. We equip each *time-series instance* with an extra time-series, monitoring the presence of missing values: in this way we can track the missing records at each time stamp.

Hence, each *time-series instance* can be described by a 6×1440 matrix, whose rows represent the type of *time-series features* (i.e., HR, ABP, pulse pressure, $\mathrm{SaO2}$, BR and missing records) and the columns the time domain. The illustration of one sample *time-series instance* is shown in Figure 2.

Missing values of vital signs signals have been imputed by using a zero-order spline, i.e., the Last Occurrence Carried Forward (LOCF) method. The inclusion of the missing values time-series helps the CNN to recognize the correct informativeness of flat patterns, i.e., whether a flat pattern is due to the LOCF method or not. We remark that our choice of 24-hour time window is only for the sake of illustrating the methodology. The analysis can be repeated with any window width (as done in Section 2 of the *Supplementary material*). However, the larger is the prediction window, the larger the dimensionality of the input data.

4.2 Design of the CNN

The CNN represents a specific class of Artificial Neural Networks (ANNs) which is designed to work with grid-structured data, e.g., time-series and images. Due to this intrinsic ability to process multi-level data, CNN have been widely applied in image recognition (Liu, 2018; Zheng et al., 2017; Lou and Shi, 2020; Kagaya et al., 2014), anomaly detection (Kwon et al., 2018; Naseer et al., 2018; Staar et al., 2019), and time-series forecasting (Borovykh et al., 2017; Selvin et al., 2017; Livieris et al., 2020; Guo-yan et al., 2019). More specifically, convolutional and max-pooling operators are combined to encode the sequentiality of the patterns contained in the input data. As a result, the optimization of the weights of the convolutional filters of the convolutional layers aims to give the most linearized latent representation of the input time-series.

In the present work we have chosen a pure convolutional network: its architecture is composed of convolutional, pooling, and dense layers only. The choice of a CNN seems natural, since we are looking for translational invariant

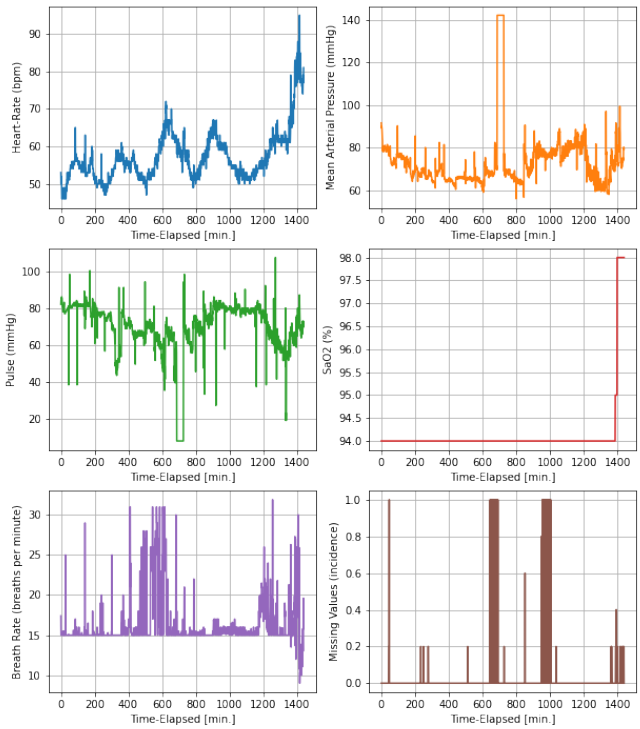


Figure 2: Example of *time-series instance*. x-axis: time-domain (24 hours). y-axis: the values taken by each time-series feature. In specific, HR in blue, ABP in orange, pulse pressure in green, SaO_2 in red, BR in purple, and the auxiliary time-series (with the missing values incidence) in brown.

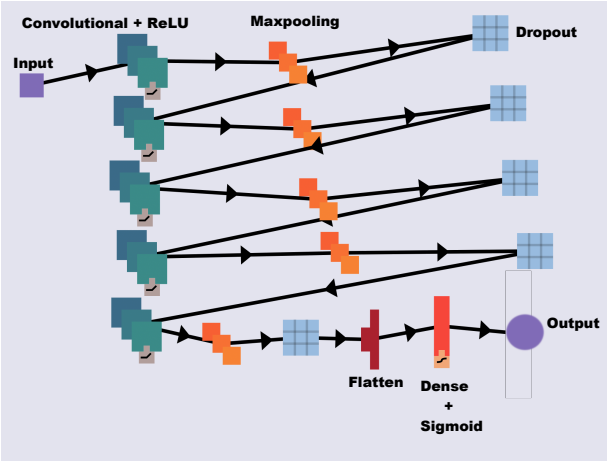


Figure 3: Schematic illustration of the CNN model. Starting from the left, the input signal is processed by a convolutional layer (128 filters of size 3). The a *ReLU* function is applied before a *max-pooling* operator that reduces the size of the features. After each *max-pooling* layer the network also contains a *dropout layer* whose dropout rate is 0.25. This sequence of hidden layers is repeated five times. The last feature map is flattened into an array and then propagated in a *fully-connected* layer (dense layer) with a sigmoid activation function.

patterns that might be present in any sub-interval in the time-series. However, in order to give quantitative grounds to this reasoning, in Section 2 of the *Supplementary Material* we compare CNN's accuracy with other traditional NN-based models, namely Logistic Regression (LR), linear Supported Vector Machine (SVM), Multi Layer Perceptron (MLP), and CNN-LSTM networks (where LSTM stands for Long Short-Term Memory). We opted for a CNN design, due to its accuracy and to the possibility of applying the saliency maps analysis, presented in Section 6.

The final architecture chosen for the CNN is the following:

1. *Convolutional Layers*: the number of filters on each layer is 128, and each filter has a size of 3 (pixels). We call a *feature map* the output of a filter applied to the previous layer.
2. *Activation Layer*: the ReLU function (i.e. $\text{ReLU}(x) := \max(0, x)$) is applied after each convolution operator. This application of a non-linear activation function on the feature maps gives rise to the *activated feature maps*.
3. *Max-pooling layer*: the activated feature maps are resampled via a max-pooling operator with a pooling size of 2 (sub-sampling).

The architecture also contains a *dropout layer* after each max-pooling layer. The dropout layer has a dropout rate of 0.25. This sequence of hidden layers is repeated five times. The last feature map is flattened into an array and then propagated into a *fully-connected* layer (dense layer) with a sigmoid activation function. The activation function returns a positive output between 0 and 1, that is, the risk score evaluated by the CNN. The architecture of the chosen CNN is sketched out in Figure 3. The figure is created by using the on-line tool *ENNUI* (https://math.mit.edu/ennui/).

4.3 Training and overall evaluation of the CNN

When training the model with the input EHR data, we used only a portion of the total amount of available EHR data. Indeed, we under-sampled the overall amount of EHR data to avoid both the training and the test set being too imbalanced. The number of *time-series instances* in the case group (i.e., those instances representing the ICU-AI episodes) are about one-twentieth of the total amount of *time-series instances* in the control group (i.e., those instances not representing the ICU-AI episodes). Thus, we fit the CNN model on a population of *time-series instances* with a control-case ratio of 8:1 (i.e., for each time-series instance in the case group one has eight time-series instances from the control group). It is important to stress that when under-sampling the EHR data, we apply a random under-sampling on the control group only. We use binary cross-entropy for the loss function, and the ADAM algorithm as the optimizer (Kingma and Ba, 2015).

Since we train the CNN to solve a binary classification task, the Area Under the Receiver Operating Characteristic curve (AUROC) score (Fawcett, 2006) represents the most appropriate choice for assessing the performance during the learning phase. Although we are not interested in the prediction formulated by the CNN in itself, we need to guarantee that the CNN model is able to classify the *time-series instances* and to encode informative patterns that describe the impending onset of an ICU-AI. Internal validation was performed using the *K-folds cross-validation* method. When validating the performance of CNN models as binary classifiers, the data were split into 5 folds. The overall AUROC is the average over the 5 folds. In Figure 3 of the *Supplementary material* the reader can find the behavior of the AUROC of the CNN model as function of three *hyper-parameters* of the network.

4.4 CNN Risk score

The extraction of the CNN score and its inclusion in the LM-CR model represent the novel ideas of the manuscript. The risk score of infection is evaluated by means of the CNN, whose architecture was discussed in Section 4.2 and its training in Section 4.3.

Thus, the procedure for evaluating the risk scores is the following:

1. Consider the vital signs signals of patient i (HR, ABP, pulse pressure, SaO2, and RR) and the missing values time-series.
2. Starting from ICU admission time, extract 24-hour *time-series instances* by means of an 8-hour sliding time window (see Section 4.1), corresponding to the intervals in $\mathcal{P}_i$.
3. Propagate the *time-series instances* through the hidden layers of the fitted CNN model and evaluate the risk-score.
4. Assign the risk score to the corresponding time-stamp (i.e., day-month-hour-minute).

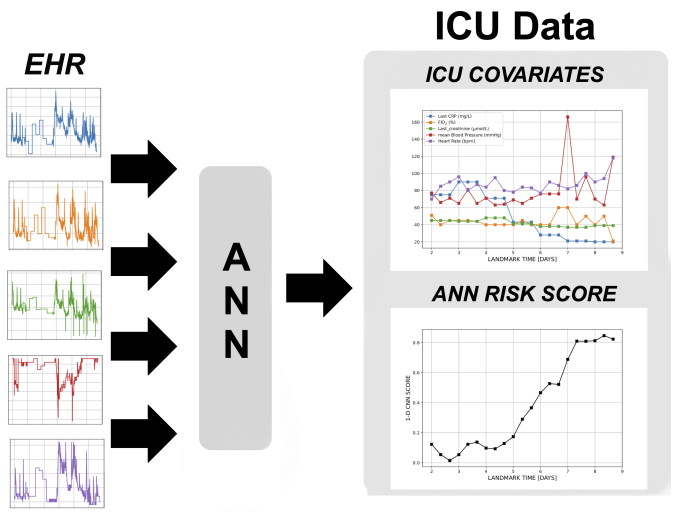


Figure 4: Schematic representation of the inclusion of the CNN-based risk score $Z_{\text{CNN}}(t_{LM})$ in the ICU cohort data.

A scheme of how we incorporated the risk score into the ICU predictors is illustrated in Figure 4: for a single patient the score is calculated for each LM time t_{LM}. At each t_{LM} the values of other time-dependent covariates are reported as well (e.g., CRP, FiO_2, creatinine level, mean blood pressure, mean heart rate).

5 Step 2: *Deep* LM-CR model

5.1 Notations and LM-CR model

In this Section, we shall present the LM model following the notation used in Nicolaie et al. (2013).

We consider a cohort consisting of N subjects, and we denote with $\tilde{T}$ the time of failure, C the censoring time, D the cause of failure, and $\mathbf{Z}(\cdot)$ and array of covariates. For the i-th subject, the tuple $(T_i, \Delta_i, \mathbf{Z}_i(\cdot))$ represents respectively the observed time $T_i = \min(\tilde{T}_i, C_i)$ (i.e., the earliest of failure and censoring time), the cause of failure $\Delta_i = \mathbf{1}(\tilde{T}_i < C_i)D_i$ (with $\mathbf{1}(\cdot)$ the indicator function), and $\mathbf{Z}_i(\cdot)$ the covariates up to time T_i. Likewise, we shall adopt the subscript j to refer to the competing causes of failure, with $j \in \{1, \dots, J\}$.

We would like to derive a dynamic prediction of the probability distribution function of the failure time of cause j at some time horizon (t_{hor}), conditional on surviving event free and on the information available at a fixed time t_{LM} (*landmark time*). More specifically, given a prediction window w (such that $t_{hor} = t_{LM} + w$) we would like to estimate the survival probability and the Cumulative Incidence Function (CIF) of cause j:

$$S_{LM}(t_{hor}|\mathbf{Z}(t_{LM}), t_{LM}) := \mathbb{P}(T > t_{hor}|\mathbf{Z}(t_{LM}), t_{LM}), \tag{1}$$

$$F_{j,LM}(t_{hor}|\mathbf{Z}(t_{LM}), t_{LM}) := \mathbb{P}(T \leq t_{hor}, \Delta = j|\mathbf{Z}(t_{LM}), t_{LM}). \tag{2}$$

The LM approach consists of two steps:

1. We first divide the time domain of our observations $[s_0, s_1]$ into n equi-spaced landmark points denoted with $\{t^k_{LM}\}^n_{k=1}$, where $t^1_{LM} \equiv s_0$ and $t^n_{LM} \equiv s_1$. Hence, we fix the width of the prediction window w (i.e., the *lead time*), and then for each LM time t^k_{LM} we create a dataset by selecting all the subjects at risk at time t^k_{LM} and by imposing *administrative* right-censoring at the time $t^k_{LM} + w$ (*horizon time*). Thus, for a vector of time-dependent covariates $\mathbf{Z}(t)$, only the values $\mathbf{Z}(t^k_{LM})$ at t^k_{LM} are considered in the k-th dataset. Finally, we create an extensive dataset by stacking all the datasets extracted at each landmark time t^k_{LM} (LM *super-dateset*).
2. The second step is fitting the *LM-CR super-model* on the stacked LM *super-dateset* (Nicolaie et al., 2013). Since at each t^k_{LM}, the vector $\mathbf{Z}(t^k_{LM})$ is treated as a time constant vector of covariates, the dataset can be analyzed by using standard survival analysis methods.

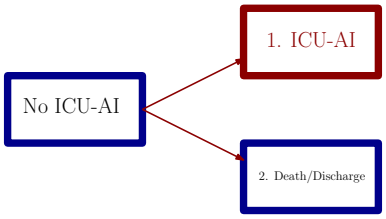


Figure 5: Competing risks model for ICU-AI.

In the *LM-CR super-model* we fit indeed a Cox proportional hazard model for the cause specific hazard λ_j:

$$\lambda_j(t|t_{LM}, \mathbf{Z}(t_{LM})) = \lambda_{0j}(t|t_{LM}) \exp\left[\beta_j^T(t_{LM})\mathbf{Z}(t_{LM})\right], \tag{3}$$

where $\lambda_{0j}(t|t_{LM})$ denotes the (unspecified) baseline hazards and $\beta_j(t_{LM})$ the set of regressors specific for the j-th cause in within the interval interval $[t_{LM}, t_{LM} + w]$. We assume that the coefficients β depend on t_{LM} in a smooth way, i.e., $\beta_j(t_{LM}) = f_j(t_{LM}, \beta_j^{(0)})$ with $\beta_j^{(0)}$ a vector of regression parameter and $f_\beta(\cdot)$ a parametric function on time, e.g., a spline. Our choice has been a quadratic function:

$$\beta_j(t_{LM}) := \beta_j^{(0)} + \beta_j^{(1)} t_{LM} + \beta_j^{(2)} t_{LM}^2$$

Fitting this model with the Breslow partial likelihood for tied observations is equivalent to maximizing the pseudo-partial log-likelihood as shown in (Nicolaie et al., 2013). The landmark supermodel can be then fitted directly by applying a simple Cox model to the stacked data set. Hence, after having estimated the coefficients and the baseline cause specific hazards, we get the *plug-in* estimators for the survival probabilities (i.e., $\hat{S}_{LM}(t_{hor}|\boldsymbol{Z}(t_{LM}), t_{LM})$) and of the CIF of cause j (i.e., $\hat{F}_{j,LM}(t_{hor}|\boldsymbol{Z}(t_{LM}), t_{LM})$).

5.2 LM-CR for ICU-AI

In the context of dynamic predictions for ICU-AIs, we adopted a CR model with three causes of failure: *ICU-AI*, *death in the ICU* and *discharge*; see Figure 5. No right censoring is present in the data, since no patient left the ICU before discharge or death.

Following the notation used in Section 5.1, we denote with $\tilde{T}$ the time of failure, D the cause of failure (i.e., $D = 1$ denotes an ICU-AI, while $D = 2$ discharge or death), and $\mathbf{Z}(\cdot)$ the array of covariates. For the i-th subject the triple $(T_i, \Delta_i, \mathbf{Z}_i(\cdot))$ denotes the observed time $T_i \equiv \tilde{T}_i$, the cause of failure $\Delta_i \equiv D_i$, and $\mathbf{Z}_i(\cdot)$ the vector of covariates.

In this article, we consider the prediction window was set to $w = 24$ hours. The time domain is $[s_0, s_1]$, with $s_0 = 48$ hours and $s_1 = 240$ hours, and we consider $n = 25$ LM times t_{LM}, i.e., two subsequent LM times are at distance 8 hrs.

If we denote with $Z_{\text{CNN}}(t)$ the CNN risk score at time t (see Figure 4) and with $\mathbf{Z}(t)$ the vector of all the other covariates in the LM-CR model at time t, we are interested at the dynamic predictions of the two models:

1. $\pi_1 := F_{1,LM}(t_{hor}|\mathbf{Z}(t_{LM}), t_{LM})$: i.e., the CIF of infection conditioned on the survival up to time t_{LM} and on the *low frequency* covariates (LM-CR model);
2. $\pi_2 := F_{1,LM}(t_{hor}|\mathbf{Z}(t_{LM}), Z_{\text{CNN}}(t_{LM}), t_{LM})$: the CIF of infection conditioned on the survival up to time t_{LM} and on both the *low frequency* covariates and Z_{CNN} (Deep-LM-CR model).

By comparing the accuracies of π_1 and π_2, we can measure the added predictive power of the CNN score. We shall refer at the first model with LM-CR and to the second with Deep-LM-CR.

5.3 Evaluation of LM-CR model

We use the AUROC metric to evaluate the prediction made at each single landmark time. When considering an overall measure, the evaluation of a global AUROC needs to consider the time-dependent character of the dynamic. Similarly to the estimator of the prediction error proposed in Spitoni et al. (2018), the evaluation of the overall AUROC needs to take into account the change in time of the size of the risk-set. The absence of censoring allows us to estimate the overall AUROC score simply by:

$$\text{AUROC}_{\text{global}} = \frac{\sum_{k=1}^{n} R(t_{LM}^k)\,\text{AUROC}(t_{LM}^k)}{\sum_{k=1}^{n} R(t_{LM}^k)}, \tag{4}$$

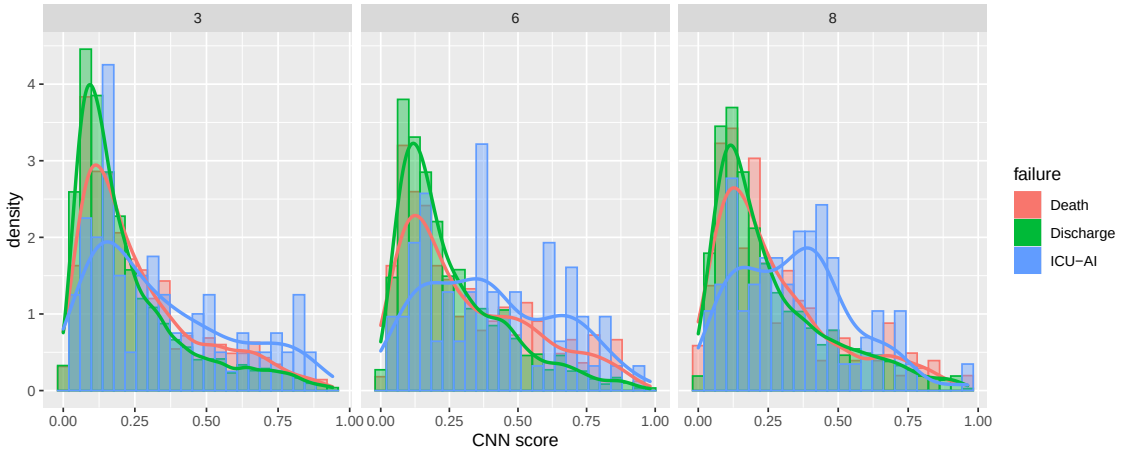


Figure 6: Distribution of the CNN risk score at three different landmark points ($t^k_{LM} \in \{3, 6, 8\}$ days), stratified for the cause of failure.

with t^k_{LM} the k-th landmark time, n the total number of landmark times, and $R(t^k_{LM})$ the size of the risk-set at time t^k_{LM}.

The influence of the individual predictor in the prediction has been visualized by means of heat-maps. We compute the relative variation of the overall AUROC between the model including all predictors and the one where the predictor is removed. Thus, we construct a heat-map representing the relative change in AUROC due the removal of a single predictor at landmarking time t_{LM}.

Finally, we remark that internal validation was performed using a *10-folds cross-validation* method. The overall $\text{AUROC}_{\text{global}}$ and the $\text{AUROC}(t^k_{LM})$, evaluated at each time t^k_{LM}, are averaged over the 10 folds. In both the CR-LM model and the Deep-CR-LM model, we report 95% bootstrap confidence intervals.

5.4 Results

In this Section we are going to show that the CNN risk score Z_{CNN} adds extra predictive information to the model, not present in the standard covariates.

In Figure 6 we plotted the empirical distribution of $Z_{\text{CNN}}(t_{LM})$ for three landmark points (i.e., $t_{LM} \in \{3, 6, 8\}$) and stratified by the cause of failure. As expected, the distribution of Z_{CNN} for infected patients is more skewed on the right: while at day three this phenomenon is mild, at days 6 and 8 the skewness of the density distribution is much more evident.

In Figure 7 we report the Pearson correlations between the CNN risk score and the vital signals averaged in the 24hrs time windows prior the landmark (time-dependent covariates included in the LM-CR). Although the risk score is evaluated relative to these signals, only mild correlations are present. Our main hypothesis is indeed that $Z_{CNN}(t_{LM})$ has added predictive information, not contained in the other covariates $\mathbf{Z}(t_{LM})$.

Moreover, with regards to the cause specific hazards for infection, the CNN risk score turned out to be the most important predictor: $\beta^{(0)}_{1;\text{CNN}} = 4.8$ (95%CI 3.05-6.72). All the cause-specific hazards for ICU-AI are reported in Table 3 of the *Supplementary Material*.

The LM approach provides a *plug-in* estimator for the dynamic prediction (2) of the CIFs of ICU-AI. Therefore, in order to give an example of the dynamic prediction allowed by the model, in Figure 8 we report the CIFs for the LM-CR and the Deep-LM-CR models as function of the landmark time and of the quantile groups of the fitted linear predictors. Given the value of the covariates at the landmark time t_{LM}, the CIF at any s, with $s \in [t_{TM}, t_{LM} + w]$ is given indeed by the *plug-in* estimator $\hat{F}_{1,LM}(s|\boldsymbol{Z}(t_{LM}), t_{LM})$ of (2). The dashed red line in Figure 8 denotes an arbitrary warning level for the CIF of infection (e.g., 8%). We can see that, for the forth quantile Q_4 and at LM time $t_{LM} = 4$ days, the Deep-LM-CR model has a *lead time* of circa 3 hours in reaching the warning threshold before the LM-CR model.

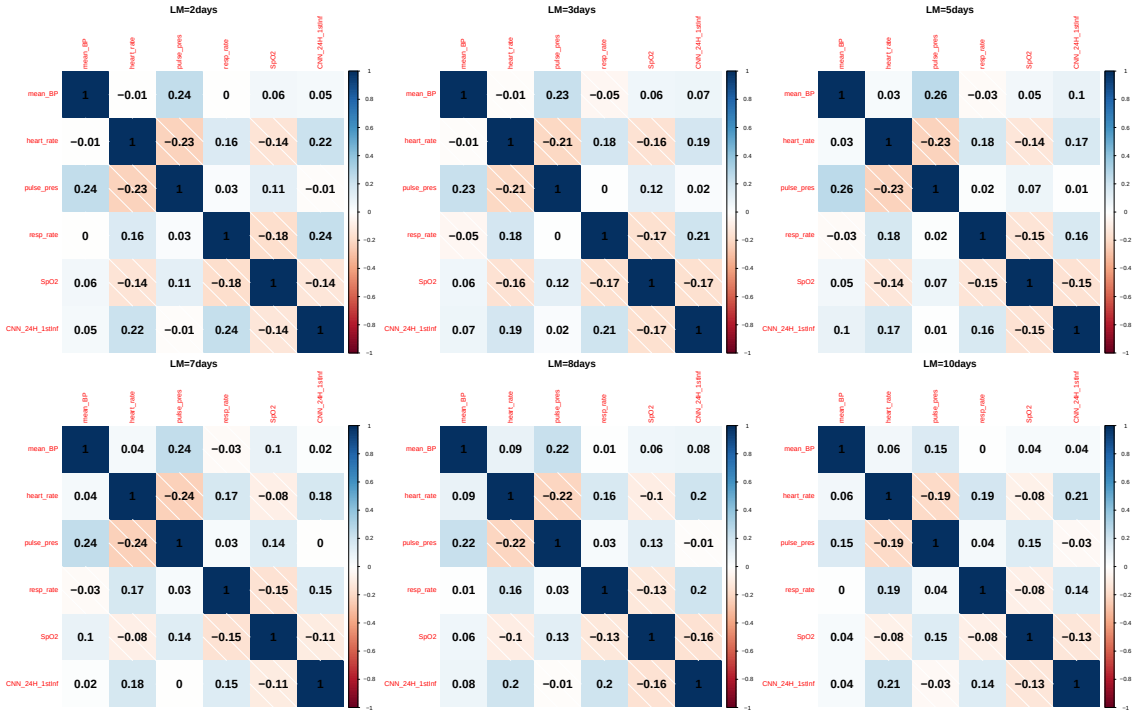


Figure 7: Correlation plot: CNN risk score vs. the vital signals (averaged in the 24 hours before the landmark).

The overall measure for the LM-CR model is $\mathrm{AUROC}_{\mathrm{global}} = 0.69$ (95%CI 0.68-0.70), while for the Deep-LM-CR is $\mathrm{AUROC}_{\mathrm{global}} = 0.75$ (95%CI 0.73-0.76). The $\mathrm{AUROC}(t^k_{LM})$ scores evaluated at each time t^k_{LM}, with $k \in \{1, \ldots, n\}$ are shown in Figure 9. The LM-CR model always shows lower predictive performance than the Deep-LM-CR. We notice that at the beginning of the ICU stay (days 3-4) and around day 7, the CNN can improve the prediction of the traditional ICU clinical covariates of about 8%, see Figure 10.

The impact of each explanatory variable Z_j involved in the Deep-LM-CR model is shown in Figure 11, whereas we reported the heat-map of the relative increase in AUROC between the Deep-LM-CR without the covariate Z_j and the full model (with $\mathbf{Z}(t_{LM})$ and $Z_{CNN}(t_{LM})$). When $Z_j = Z_{CNN}$, we see that we observe a relative increase in AUROC of at least 4%.

Summing up, we have shown that the two-step modeling can effectively lead to an increase of the accuracy of the predictions. The extra predicting power comes from the inclusion of the CNN-based risk score, which is a summary measure of the predicting patterns found by the CNN model trained on only five vital signs signals (sample frequency of 1 minute).

We remark that in our analysis we did not consider recurrent infections, but we limited the attention to the first episode of ICU-AI.

6 Explainability of CNN-based prediction of ICU-AI

In this section, we present an attempt to make interpretable the activity of the CNN. As shown in Section 5.4, the CNN-based risk score has added predicting power to the LM-CR model. However, for the moment, we do not have any information about the saliency of the vital signs signals selected by the CNN during the training. This knowledge might be crucial for shedding some light on the relation between the activity of pattern recognition of the network and the medical conditions of a patient when a ICU-AI is approaching.

To investigate which characteristics of the pattern selected by the CNN, we use the so-called *Explainable Artificial Intelligence* (XAI), namely a class of methods designed to understand the decisions and the predictions formulated by ANN techniques (Phillips et al., 2020; Vilone and Longo, 2021; Castelvecchi, 2016). The scope of XAI is to contrast indeed the widespread *black box* attitude that many users have when applying ANN techniques.

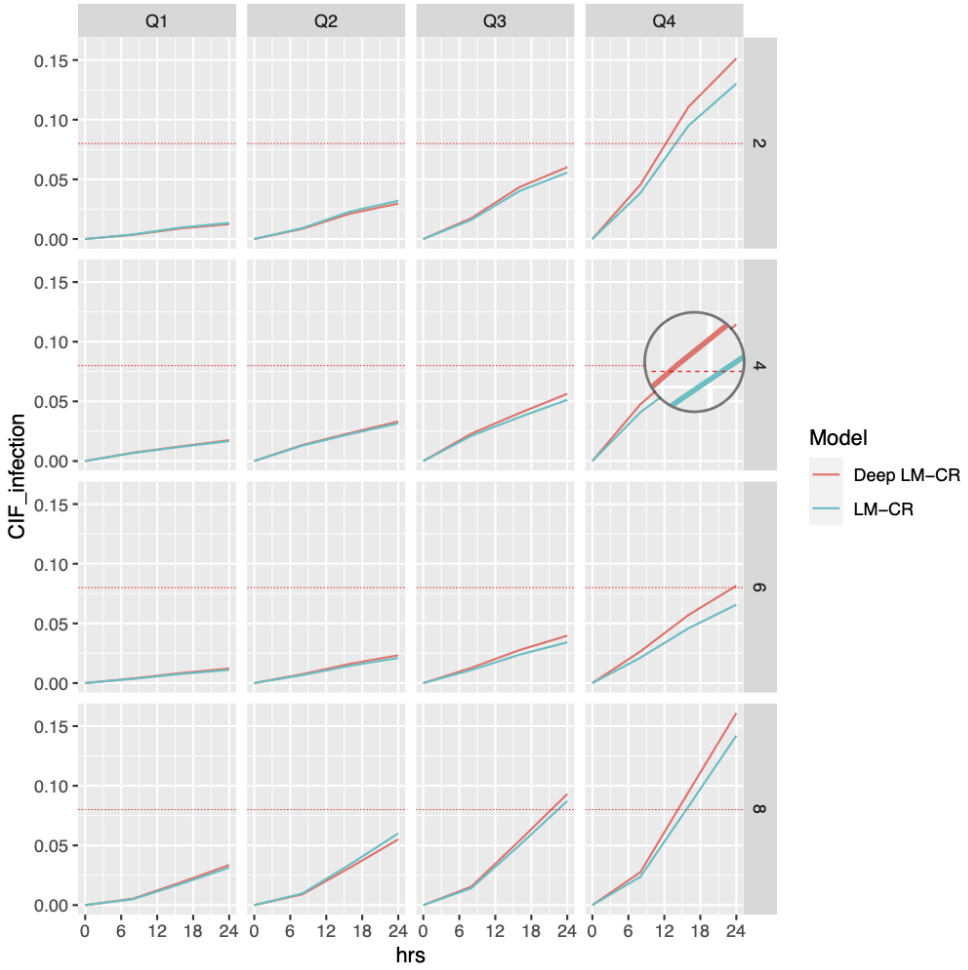


Figure 8: Comparison of the CIFs at different landmark time (i.e., $t_{LM}^k \in \{2, 4, 6, 8\}$ days) of the models LM-CR and Deep-LM-CR.

6.1 Explanability via SMOE scale

A saliency map is a map acting on the activated features in the hidden layers, generally used for showing which parts of the input are most important for the network's decisions. The Saliency Map Order Equivalent scale (SMOE) used in the present paper is base on the algorithm developed by Mundhenk et al. (2019): an efficient and non-gradient method based on the statistical analysis of the activated feature maps. For a more detailed description of the SMOE scale, we refer the reader to Section 3 of the *Supplementary Material*.

We would like to use the saliency maps for selecting, in the original 24hrs time-series, the most relevant 8-hours patterns.

The adopted approach is the following:

1. We fit three different CNNs, one for each of $t_{LM}^k \in \{3, 7, 10\}$. We consider three distinct CNNs because the predicting patterns found by the network might differ among different periods of the ICU stay (see for instance the discussion in Section 6.3). The LM point *3 days* is a proxy for an early time of the stay, *7 days* for an intermediate time, and finally *10 days* for a later moment. The design of the networks is the same as described in Section 4.2. All these models are validated via 5-fold cross-validation.
2. We study the pattern recognition performed by the hidden layer, and we make it interpretable via the *SMOE scale*. Through this method, we can visualize the regions of the input data with the highest saliency. Specifically, for each model developed at every LM time t_{LM}^k, we construct and visualize the saliency maps of the test set only. We repeat this action for each test set of each cross-validation fold.

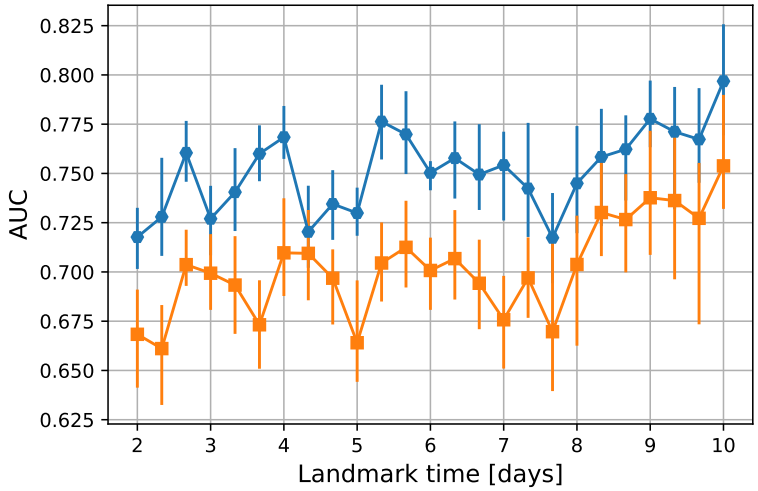


Figure 9: AUROC score (y-axis) as a function of the landmark times (x-axis). The two curves represent the predictive performance of the basic CR-LM model (orange), and of the Deep-CR-LM model (blue) The error bars denote the 95% bootstrap confidence intervals.

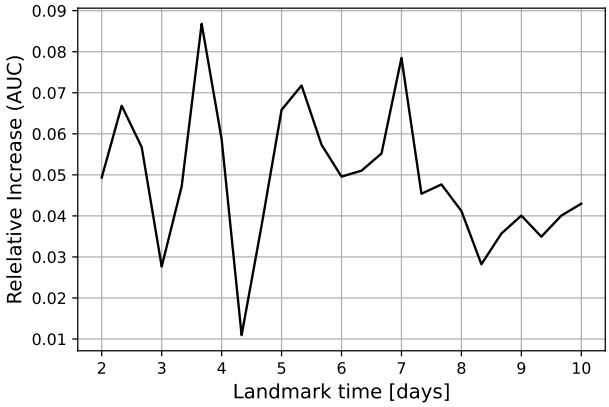


Figure 10: ICU-AI: overall relative increase of AUROC score (y-axis) as a function of the landmark times (x-axis) when including CNN-based risk score.

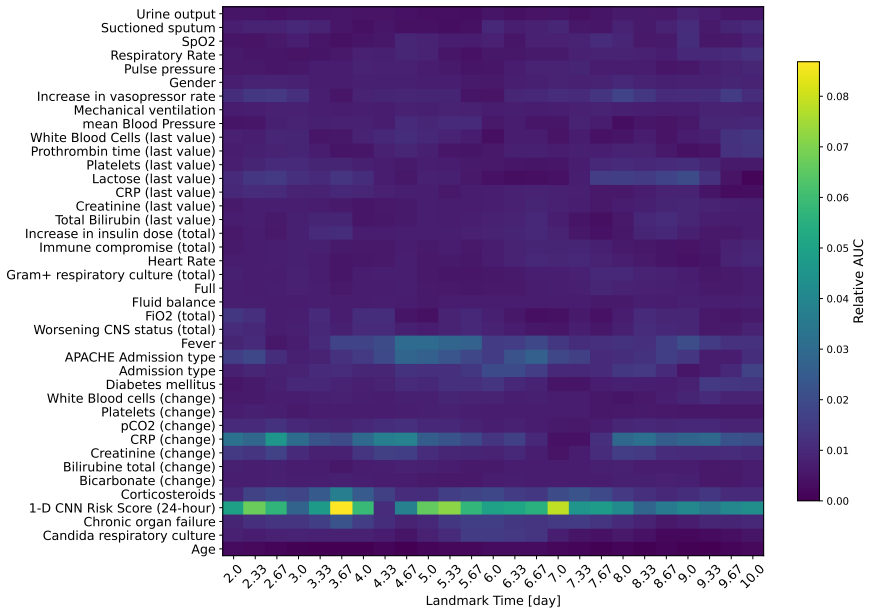


Figure 11: AUC heat-maps evaluating the impact of each predictor in the Deep-LM-CR model when predicting ICU-AI. The color of each pixel denotes the magnitude of the impact (relative AUROC increase) of one covariate (y-axis) with respect to the LM time (x-axis).

3. From each saliency map, we extract the *8-hours interval* with the highest *cumulative saliency value*. After having extracted the most relevant 8-hour patterns from each *time-series instance*, we can focus on their interpretation and their clustering. An example of the extraction of the 8-hours most salient pattern is shown in Figure 12.

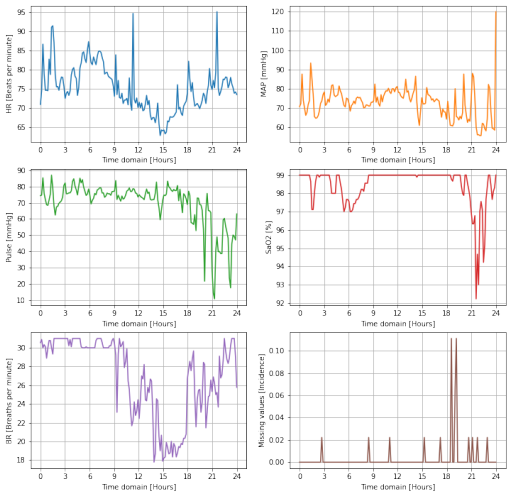
6.2 Data-driven clustering of salient patterns

We focus now the attention on the clustering of the most salient patters extracted in Section 6.1. We would like indeed to answer the question: *how can we link the activity of pattern recognition to some medical conditions, appearing when a ICU-AI is approaching?* Our strategy for answering the question is the following:

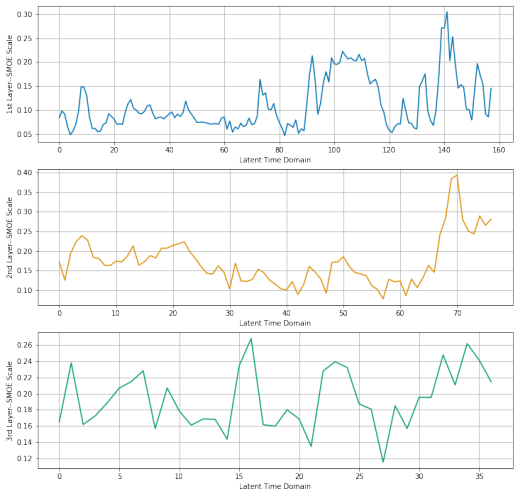
1. We collect the set of the most predictive patterns with amplitude 8 hours, obtained by applying the SMOE scale to the time-series instances, as explained in Section 6.1.
2. We consider four clinical critical conditions, i.e., *tachycardia*, *hypotension*, *desaturation*, and *hyperventilation* (see Table 1), which could predict the approaching of one ICU-AI episode. These medical conditions reflect the main symptoms of the Systemic Inflammatory Response Syndrome (SIRS), see Chakraborty and Burns (2019). Tachycardia, hypotension, and hyperventilation are quite spread in the ICU, and they usually mentioned in general guidelines for the ascertainment of SIRS (Comstedt et al., 2009). For the criteria reported in Table 1 we refer to Comstedt et al. (2009); in specific for Desaturation, we refer to (Hafen and Sharma, 2022).
3. We evaluate the mean values of HR, ABP, SaO_2 and BR for each of the most salient *8-hour pattern* extracted via the SMOE scale. Depending on the values obtained (see the criteria in Table 1), we check the presence of the four clinical critical conditions. Thus, the combination of these conditions produces 16 different possible clinical situations of interest, as shown in Table 2: they represent the classes of the proposed data-driven clustering. In Figure 13 the 16 distinct classes are represented as nodes of a graph (i.e., a four dimensional hypercube).

6.3 Results of the data-driven clustering

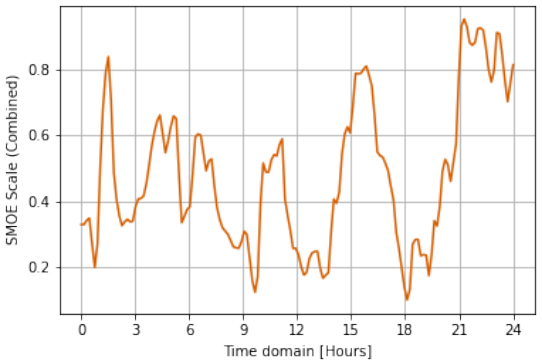
Histograms with the relative frequencies of the 16 data-driven clusters are shown in Figure 14. For day 3 (see Figures 14(a) and 14(b)), a two-sample Kolmogorov-Smirnov test (Hodges, 1958) reveals that the sample distributions



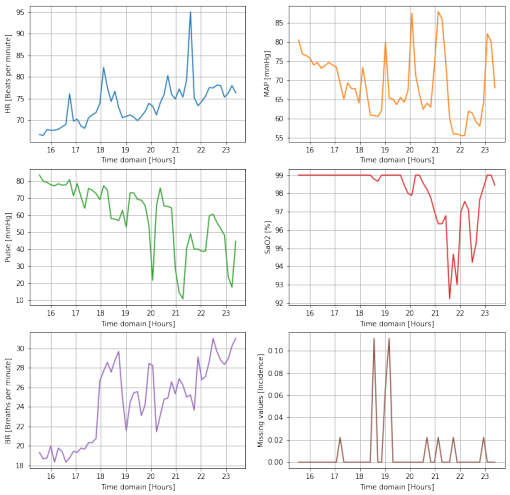
(a)



(b)



(c)



(d)

Figure 12: Schematic example of the extraction of the most salient patterns in the 24-hours time-series instances: (a) Example of time-series instance, (b) SMOE scale applied on each activation feature map of the CNN, (c) Averaged saliency map (weighted average of SMOE scales on individual hidden layers), (d) extraction of the most salient interval of (a).

Critical Condition	Criterion
Tachycardia	Hearth Rate $\geq$90 beats per minute
Hypotension	Arterial Blood Pressure (mean) $\leq$ 80mmHg
Desaturation	$SaO_2 \leq 95\%$
Hyperventilation	Breath Rate $\geq$ 24 breaths per minute

Table 1: Critical conditions and their criteria.

Class	Data Driven Cluster (Clinical Conditions)
0	None
1	Tachycardia
2	Hypotension
3	Hypotension, Tachycardia
4	Desaturation
5	Desaturation, Tachycardia
6	Desaturation, Hypotension
7	Desaturation, Hypotension, Tachycardia
8	Hyperventilation
9	Hyperventilation, Tachycardia
10	Hyperventilation, Hypotension
11	Hyperventilation, Hypotension, Tachycardia
12	Hyperventilation, Desaturation
13	Hyperventilation, Desaturation, Tachycardia
14	Hyperventilation, Desaturation, Hypotension
15	Hyperventilation, Desaturation, Hypotension, Tachycardia

Table 2: List of the 16 clinical conditions (classes of the clustering).

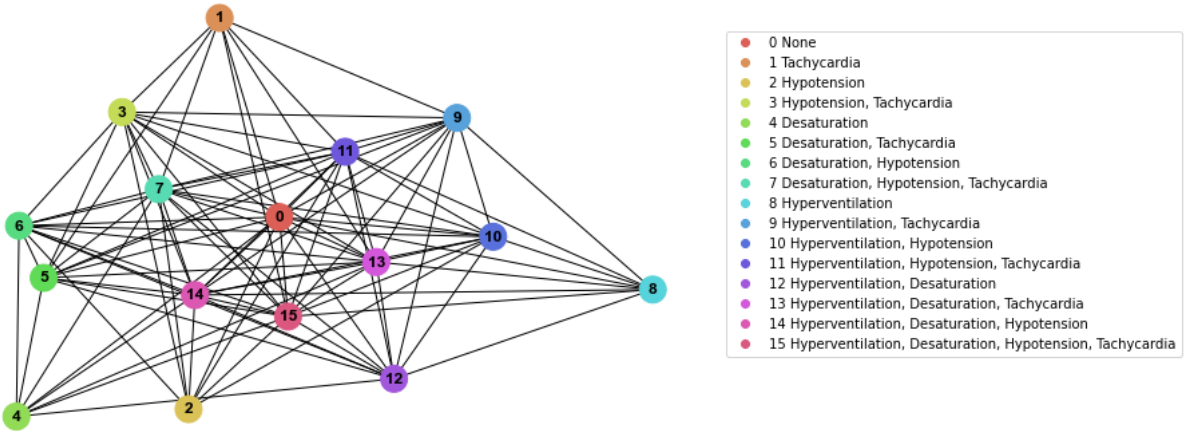
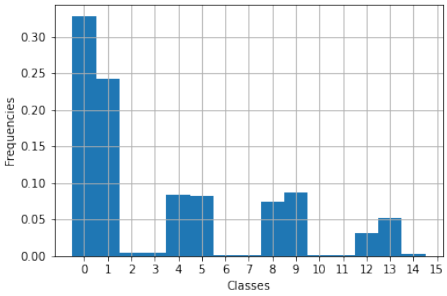


Figure 13: Graph whose nodes represent the 16 classes of the clustering.

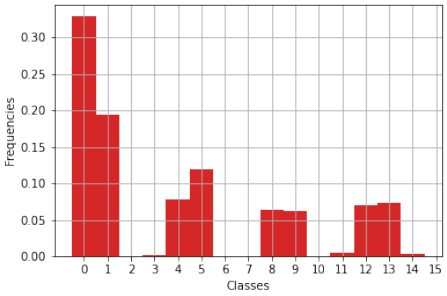
of the classes between *not-infected* and *infected* instances are not significantly different (p-value=0.21). However, we can observe a completely different scenario on both days 7 and 10 (see Figure 14 (d)-(f)), where the null hypothesis of the two-samples Kolmorogov-Smirnov test is rejected (p-value= 0.0003 and p-value= 10^{-10} respectively). Hence, this analysis shows that different clinical conditions could represent an essential feature of the patterns that the CNN model captures during the learning phase. For instance, for *infected* instances, at day 10, the prevalence of at least one of these 16 conditions is around 94%, while 79% at day 7; see Figure 14 (d)-(f)). Precisely, on day 10, events with hyperventilation correspond at 70% of samples, and in combination with tachycardia 23%. While a day 7 tachycardia is much more relevant and occurs in 50% of infectious samples. Therefore, the most salient 8-hours subinterval of our *time-series instance* can be linked to precise medical conditions, which are known to be related to the presence of an ICU-AI.

7 Conclusions

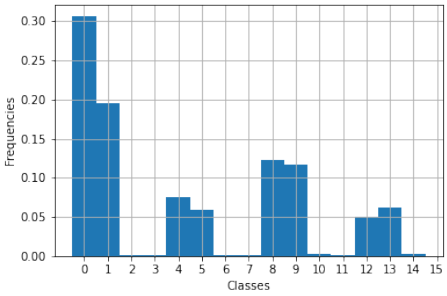
We have showed that the proposed two-step modeling of ICU-AI is at the same time an accurate predicting tool and an interpretable model. The CNN is able to detect predicting patterns by analyzing the time-series of five vital sign signals. These patters contain extra predictive information and they are only mildly correlated with the averaged



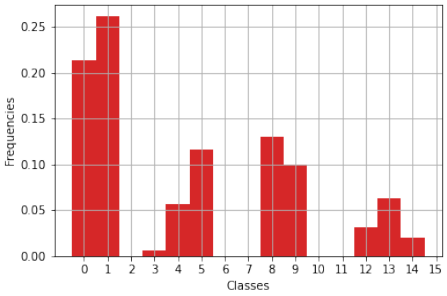
(a)



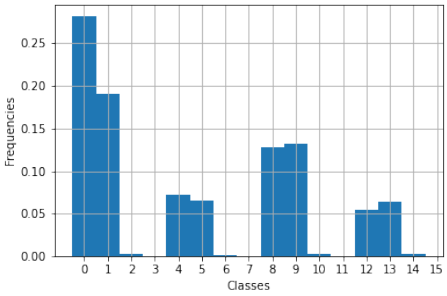
(b)



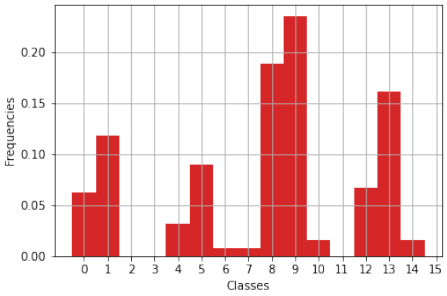
(c)



(d)



(e)



(f)

Figure 14: Histograms the data-driven clustering approach. Bins on the x-axis represent the 16 classes. Blue histograms concern the non-infected instances, whereas the red ones the infected instances. CNN trained on day 3 is described by (a) and (b), on day 7 by (c) and (d), and on day 10 by (e) and (f).

quantities of the vital signals, routinely included in the traditional survival models. Moreover, we have showed as well that the SMOE scale might help physicians in clustering patients with an approaching infection.

We have illustrated the methodology in a competing risks framework. However, recently the LM approach has been extended to *multi-state* models, even without the Markov assumption (Putter and Spitoni, 2018; Hoff et al., 2019). Therefore, as a further extension we could model recurrent infections as new states in a non-Markov multi state model, with transition hazards that might depend indeed on the previous infections' sequence. Moreover, another future challenging direction of investigation is a sort of *inversion* of the CNN, in order to identify and classify the patterns in the signal with higher predicting power. This analysis might help in performing a more precise clustering of the patients with fore-coming ICU-AI.

Code Availability

Python codes and modules are available on GitHub: the reader can refer to `https://github.com/glancia93/ICUAI-dynamic-prediction/blob/main/ICUAI_module.py`.

References

Andersen, P. K., Borgan, O., Gill, R. D., and Keiding, N. (1993). *Statistical Models Based on Counting Processes*. Springer.

Borovykh, A., Bohte, S., and Oosterlee, K. (2017). Conditional time series forecasting with convolutional neural networks. In *Lecture Notes in Computer Science/Lecture Notes in Artificial Intelligence*, pages 729–730.

Castelvecchi, D. (2016). Can we open the black box of ai? *Nature News*, 538(7623):20.

Chakraborty, R. K. and Burns, B. (2019). Systemic inflammatory response syndrome.

Comstedt, P., Storgaard, M., and Lassen, A. T. (2009). The systemic inflammatory response syndrome (sirs) in acutely hospitalised medical patients: a cohort study. *Scandinavian journal of trauma, resuscitation and emergency medicine*, 17(1):1–6.

Cortese, G. and Andersen, P. K. (2010). Competing risks and time-dependent covariates. *Biometrical Journal*, 52(1):138–158.

Dantes, R. B. and Epstein, L. (2018). Combatting sepsis: a public health perspective. *Clinical infectious diseases*, 67(8):1300–1302.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874. ROC Analysis in Pattern Recognition.

Ferrer, L., Putter, H., and Proust-Lima, C. (2019). Individual dynamic predictions using landmarking and joint modeling: Validation of estimators and robustness assessment. *Statistical Methods in Medical Research*, 28(12):3649–3666.

Guo-yan, X., Jin, Z., Cun-you, S., Wen-bin, H., and Fan, L. (2019). Combined hydrological time series forecasting model based on cnn and mc. *Computer and Modernization*, (11):23.

Hafen, B. B. and Sharma, S. (2022). *Oxygen saturation*. StatPearls, StatPearls Publishing.

Hodges, J. L. (1958). The significance probability of the smirnov two-sample test. *Arkiv för Matematik*, 3(5):469–486.

Hoff, R., Putter, H., Mehlum, I. S., and Gran, J. M. (2019). Landmark estimation of transition probabilities in non-markov multi-state models with covariates. *Lifetime Data Analysis*, 25(4):660–680.

Ivanov, O., Molander, K., Dunne, R., Liu, S., Masek, K., Lewis, E., Wolf, L., Travers, D., Brecher, D., Delaney, D., et al. (2022). Accurate detection of sepsis at ed triage using machine learning with clinical natural language processing. *arXiv preprint arXiv:2204.07657*.

Kagaya, H., Aizawa, K., and Ogawa, M. (2014). Food detection and recognition using convolutional neural network. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 1085–1088.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *ICLR*.

Klouwenberg, P. M. K., Ong, D. S., Bos, L. D., de Beer, F. M., van Hooijdonk, R. T., Huson, M. A., Straat, M., van Vught, L. A., Wieske, L., Horn, J., et al. (2013). Interobserver agreement of centers for disease control and prevention criteria for classifying infections in critically ill patients. *Critical care medicine*, 41(10):2373–2378.

Kwon, D., Natarajan, K., Suh, S. C., Kim, H., and Kim, J. (2018). An empirical study on network anomaly detection using convolutional neural networks. In *ICDCS*, pages 1595–1598.

Liu, Y. H. (2018). Feature extraction and image recognition with convolutional neural networks. In *Journal of Physics: Conference Series*, volume 1087, page 062032. IOP Publishing.

Livieris, I. E., Pintelas, E., and Pintelas, P. (2020). A cnn–lstm model for gold price time-series forecasting. *Neural computing and applications*, 32(23):17351–17360.

Lou, G. and Shi, H. (2020). Face image recognition based on convolutional neural network. *China Communications*, 17(2):117–124.

Maki, D. G., Crnich, C. J., and Safdar, N. (2008). Nosocomial infection in the intensive care unit. *Critical care medicine*, page 1003.

May, R., Dandy, G., and Maier, H. (2011). Review of input variable selection methods for artificial neural networks. In Suzuki, K., editor, *Artificial Neural Networks*, chapter 2. IntechOpen, Rijeka.

Mundhenk, T. N., Chen, B. Y., and Friedland, G. (2019). Efficient saliency maps for explainable ai. *arXiv preprint arXiv:1911.11293*.

Naseer, S., Saleem, Y., Khalid, S., Bashir, M. K., Han, J., Iqbal, M. M., and Han, K. (2018). Enhanced network anomaly detection based on deep neural networks. *IEEE access*, 6:48231–48246.

Nicolaie, M., Van Houwelingen, J., De Witte, T., and Putter, H. (2013). Dynamic prediction by landmarking in competing risks. *Statistics in medicine*, 32(12):2031–2047.

Phillips, P. J., Hahn, C. A., Fontana, P. C., Broniatowski, D. A., and Przybocki, M. A. (2020). Four principles of explainable artificial intelligence. *Gaithersburg, Maryland*.

Proust-Lima, C. and Taylor, J. M. (2009). Development and validation of a dynamic prognostic tool for prostate cancer recurrence using repeated measures of posttreatment psa: a joint modeling approach. *Biostatistics*, 10(3):535–549.

Putter, H. and Spitoni, C. (2018). Non-parametric estimation of transition probabilities in non-markov multi-state models: The landmark aalen–johansen estimator. *Statistical Methods in Medical Research*, 27(7):2081–2092.

Rizopoulos, D. (2011). Dynamic predictions and prospective accuracy in joint models for longitudinal and time-to-event data. *Biometrics*, 67(3):819–829.

Rizopoulos, D. (2012). *Joint models for longitudinal and time-to-event data: With applications in R*. CRC press.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E., Menon, V. K., and Soman, K. (2017). Stock price prediction using lstm, rnn and cnn-sliding window model. In *2017 international conference on advances in computing, communications and informatics (icacci)*, pages 1643–1647. IEEE.

Spitoni, C., Lammens, V., and Putter, H. (2018). Prediction errors for state occupation and transition probabilities in multi-state models. *Biometrical Journal*, 60(1):34–48.

Staar, B., Lütjen, M., and Freitag, M. (2019). Anomaly detection with convolutional neural networks for industrial surface inspection. *Procedia CIRP*, 79:484–489.

Topol, E. J. (2019). High-performance medicine: the convergence of human and artificial intelligence. *Nature medicine*, 25(1):44–56.

van Houwelingen, H. and Putter, H. (2011). *Dynamic prediction in clinical survival analysis*. CRC Press.

Van Houwelingen, H. C. (2007). Dynamic prediction by landmarking in event history analysis. *Scandinavian Journal of Statistics*, 34(1):70–85.

Vilone, G. and Longo, L. (2021). Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76:89–106.

Vincent, J., Rello, J., and Marshall, J. (2009). International study of the prevalence and outcomes of infection in intensive care units. *JAMA*, 302(21):2323–9.

Zeng, Z., Hou, Z., Li, T., Deng, L., Hou, J., Huang, X., Li, J., Sun, M., Wang, Y., Wu, Q., et al. (2022). A deep learning approach to predicting ventilator parameters for mechanically ventilated septic patients. *arXiv preprint arXiv:2202.10921*.

Zheng, H., Fu, J., Mei, T., and Luo, J. (2017). Learning multi-attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 5209–5217.

Supplementary Material

1 Data, covariates and hazards

This study was conducted within the framework of the Molecular Diagnosis and Risk Stratification of Sepsis (MARS) study (ClinicalTrials.gov identifier NCT01905033), a prospective ICU cohort, for which the institutional review board approved an opt-out method of informed consent (protocol number 10-056C). Time-fixed variables included in the model are reported in Table 1, while time-dependent covariates are listed in Table 2.

Variable name	**Variable description**
Sex	Sex (male/female)
Age	Age at ICU admission
Immunodeficiency	Immunocompromised status; defined as having acquired immune deficiency syndrome, the use of corticosteroids in high doses (equivalent to prednisolone of >75 mg/day for at least 1 week), current use of immunosuppressive drugs, current use of antineoplastic drugs, recent hematologic malignancy, or documented humoral or cellular deficiency
Readmission	Previous ICU admission during current hospitalization period
Primary specialty	Diagnostic category of ICU admission (cardiovascular, gastrointestinal, neurological, respiratory, post-transplantation, trauma, other)
Diabetes Mellitus	Medical history of diabetes mellitus
Chronic corticosteroid use	Chronic medication use: systemic corticosteroids
Chronic organ failure	Presence of chronic organ insufficiency with one of the following conditions documented in medical history: • Chronic heart failure defined as the medical history of chronic NYHA class 2-4 or documented ejection fraction <45% (on echography in 2 years prior to ICU admission) or orthopnea with chronic diuretic use • Severe cardiovascular insufficiency defined as angina or dyspnea in rest or during minimal exercise (NYHA IV) • Chronic renal insufficiency defined as chronically elevated serum creatinine >177 μmol/L or chronic dialysis • Chronic restrictive, obstructive or vascular pulmonary disease leading to severe functional impairment • Chronic liver failure with portal hypertension (with positive liver biopsy) and/or upper gastrointestinal bleeding due to portal hypertension and/or episode of hepatic encephalopathy/coma due to medical history of liver failure
Admission type	Admission to a medical/surgical tertiary ICU

Table 1: Table with all the baseline predictors.

The cause-specific hazards for infection $\beta_1^{(0)}$ of the fitted Deep LM-CR model are shown in Table 3.

2 ANN model selection

In this section we find the best ANN design for two different prediction windows: 24 hours and 48 hours.

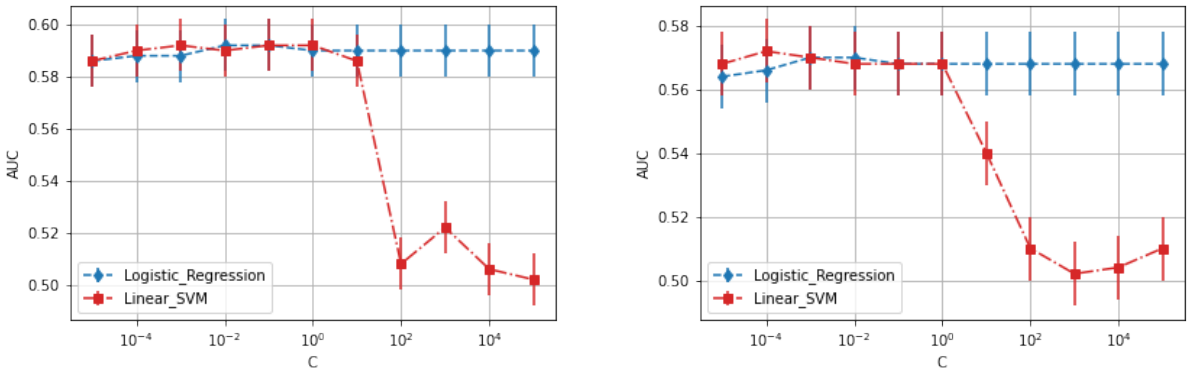
When testing the level of accuracy of LR, SVM and MLP, we need to aggregate the time-series in a suitable way. We first extract simple statistics of the time-series instances (i.e., mean value, standard deviation, skewness, kurtosis, minimum, and maximum value) on each of the physiological vital signals in a time window of 24 hours (or 48 hours). In this way we obtain a total of 31 input features. It is important to mention that the input features extracted have been linearly rescaled, in order to set the mean value and the standard deviation equal respectively to zero and one.

Variable name	**Variable description**
Heart rate	Median of 1-hour mean heart rate (bpm)
Blood pressure	Median of 1-hour mean blood pressure, either invasive mean arterial blood pressure measurement or non-invasive cuff (mmHg)
Oxygen saturation	Median of 1-hour mean oxygen saturation (%)
Respiratory rate	Median of 1-hour mean respiratory rate (rpm)
Pulse	Median of 1-hour mean pulse pressure (difference between systolic and diastolic blood pressure, mmHg)
Invasive mechanical ventilation	Last observed mechanical ventilation status
FiO2	Last observed FiO2 (inspired oxygen concentration) value in 8 hours
Chronic corticosteroid use	Chronic medication use: systemic corticosteroids
Fever	Presence of fever in last 8 hours (>38 degrees Celsius)
Fluid balance	Fluid balance (mL) over the past 8 hours
Urine output	Total urine output (mL) in 8-hour window
Suctioned sputum	Total number of times sputum was suctioned and observed within an 8-hour time window
Worsening CNS status	Either decrease in consciousness (either a decrease in GSC M-score or worsening RASS score) or onset of new delirium episode in the past 8 hours
CRP (last value)	Last observed CRP (mg/L)
CRP (change)	Unit change in CRP relative to CRP 24 hours earlier (mg/L)
White blood cell count (last value)	Last observed white blood cell count (x10^9/L)
White blood cell count (change)	Unit change in white blood cell (WBC) count relative to WBC hours earlier (x10^9/L)
Platelet count (last value)	Last observed platelet count (x10^9/L)
Platelet count (change)	Unit change in platelet count relative to platelet count 24 hours earlier (x10^9/L)
Prothrombin time (last value)	Last observed prothrombin time (seconds)
Creatinine (last value)	Last observed creatinine (μmol/L)
Creatinine (change)	Unit change in creatinine relative to creatinine 24 hours earlier (μmol/L)
Total bilirubin (last value)	Last observed total bilirubin (μmol/L)
Total bilirubin (change)	Unit change in total bilirubin relative to bilirubin 24 hours earlier (μmol/L)
Bicarbonate (change)	Unit change of bicarbonate relative to bicarbonate 24 hours earlier (mmol/L)
Lactate (last value)	Last observed lactate (mmol/L)
Increase in vasopressor rate	Increase in mean norepinephrine dose relative to previous 8-h window
Increase in insulin dose	Increase in mean insulin dose relative to previous 8-h window
Gram+ in respiratory culture	Gram-positive bacteria cultured in the airway (the result of the most recent culture)
Candida in respiratory culture	Candida species cultured in the airway (the result of the most recent culture)

Table 2: Table with all the time-dependent predictors.

Covariate	$\beta_1^{(0)}$	β-CI
Urine output	1	0.99-1.01
Suctioned sputum	1	0.99-1.02
SpO2	0.97	0.95-0.98
Respiratory Rate	1	0.99-1.01
Readmission	0.92	0.85-1.11
Pulse pressure	1	0.99-1.01
Gender (male)	1.4	1.20-1.53
Increase in vasopressor rate	1.4	1.25-1.49
Mechanical ventilation	1	0.88-1.21
mean Blood Pressure	1	0.99-1.01
White Blood Cells (last value)	1	0.99-1.01
Prothrombin time (last value)	1	0.99-1.01
Platelets (last value)	1	0.99-1.01
Lactose (last value)	0.98	0.95-1.04
CRP (last value)	1	0.99-1.01
Bilirubin (total)	1	0.99-1.01
Increase in insulin dose (total)	1	0.99-1.01
Immune compromise (total)	1.2	1.01-1.54
Heart Rate (total)	1	0.95-1.05
Gram+ respiratory culture (total)	1.1	1.01-1.54
Fluid balance	1	0.99-1.01
FiO2 (total)	1	0.99-1.01
Worsening CNS status (total)	1.1	1.02-1.18
Fever	2	1.86-2.75
APACHE-Trauma	1	0.92-1.34
APACHE-Transp	0.97	0.86-1.26
APACHE-Respir	0.67	0.54-0.77
APACHE-Other	0.58	0.33-0.96
APACHE-Neuro	1.2	1.01-1.48
APACHE-Gastro	0.71	0.54-0.96
Admission Type (surgical)	1.3	1.16-1.44
Diabetes mellitus	0.82	0.73-0.99
White Blood cells (change)	1	0.99-1.01
Platelets (change)	1	0.99-1.01
pCO2 (change)	1	0.99-1.01
CRP (change)	1	0.99-1.01
Bilirubine total (change)	1	0.99-1.01
Bicarbonate (change)	1	0.99-1.01
Corticosteroids	1.2	0.99-1.48
CNN Risk Score	4.8	3.05-6.72
Chronic organ failure	1.1	0.95-1.28
Candida respiratory culture	0.82	0.72-0.91
age	1	0.99-1.01

Table 3: Cause-specific hazards of ICU-AI



(a) (b)

Figure 1: AUROC as a function of the inverse shrinkage parameter for (a) the 24-hour and (b) the 48-hour prediction model. The red curve concerns the SVM model, while the blue line the LR model.

Both the LR and the SVM model are penalized with the L^2 norm of the weights. We use the inverse of the shrinkage parameter (here denoted as C) as the unique hyper-parameter of these two models; we then search for the best C that optimizes the AUROC score; see Figure 1. In this case, we observe that both models cannot achieve an AUROC score larger than 0.59. The 24-hour and the 48-hour prediction models present similar results.

As regards the MLP, the best accuracy of the model is intimately connected with the search of the best set of hyper-parameters. Differently from the LR and the SVM, we have more parameters to tune: *number of units, deepness, dropout rate, learning rate, activation function,* and the *batch size*. The tuning of these hyper-parameters has been done by maximizing the AUROC score. The optimization was performed over a fine grid of hyper-parameters. The MLP model was designed to optimize the binary cross-entropy by means of the ADAM optimizer. Note that the depiction of the curve of the AUROC score as a function of the hyper-parameters is quite impractical, because of the large number of parameters we had to tune. Anyway, we regarded all the configurations that meet one precise constraint (e.g., we consider all configurations with deepness equal to 3 or a number of units equal to 16) and then we selected that one with the highest AUROC score. The representation of the maximal AUROC scores should help to visualize the variation in AUROC with respect to one single hyper-parameter. In Figure 2a and 2b are shown the MLP models with a ReLU activation function (i.e., $\mathrm{ReLU}(x) = \max(0, x)$): the model does no achieve an AUROC score higher than 0.63. Similarly, the choice of a hyperbolic tangent (tanh) activation function presents a similar result; see Figure 2c and 2d.

The next model that we analyze is the CNN. Similarly to the MLP, we need to optimize over a set of hyper-parameters: the *number of convolutional filters, kernel size, deepness, learning rate*, and *the batch size*. In this case, the activation function has not been included in the hyper-parameters to tune; unlike the MLP model, we only considered the activation function ReLU. We motivate this choice after noting that several tests with a *one held-out* approach revealed that sigmoidal activation functions (e.g., sigmoid or hyperbolic tangent) affected the predictiveness of the model; one obtained AUROC always lower than 0.60 for different combinations of *power* (i.e., the number of filters $times$ dropout rate), *deepness* (i.e., number of hidden layers), and *receptive field* (i.e., the combination of kernel and max-pooling layers of different size).

Before propagating the vital signs through the CNN model, a few pre-processing steps must be performed. Firstly, one linear transformation was applied to all the instances to give a compact representation in the range [-1, 1]. Note that we applied the same-type transformation to all instances; according to the time-series feature to rescale, a precise linear transformation was applied. For example, we used the same linear transformation to rescale all heart rate signals contained in all instances; but for all breath rate signals, we developed and used a different one. Hence, for each time-series feature, we constructed a linear map that rescales both maximum and minimum values to 1 and -1, respectively. For example, if we consider the heart rate predictor in the time-series instances, we know that the minimum and the maximum value are 41 and 239 beats per minute, respectively. Accordingly, if we denote with $X_i^{HR}(t)$ the heart rate feature of the i-th time series instance, the transformation we shall apply is

$$X_i^{HR}(t) \to \frac{2X_i^{HR}(t) - 41\text{bpm} - 239\text{bpm}}{239\text{bpm} - 41\text{bpm}}.$$

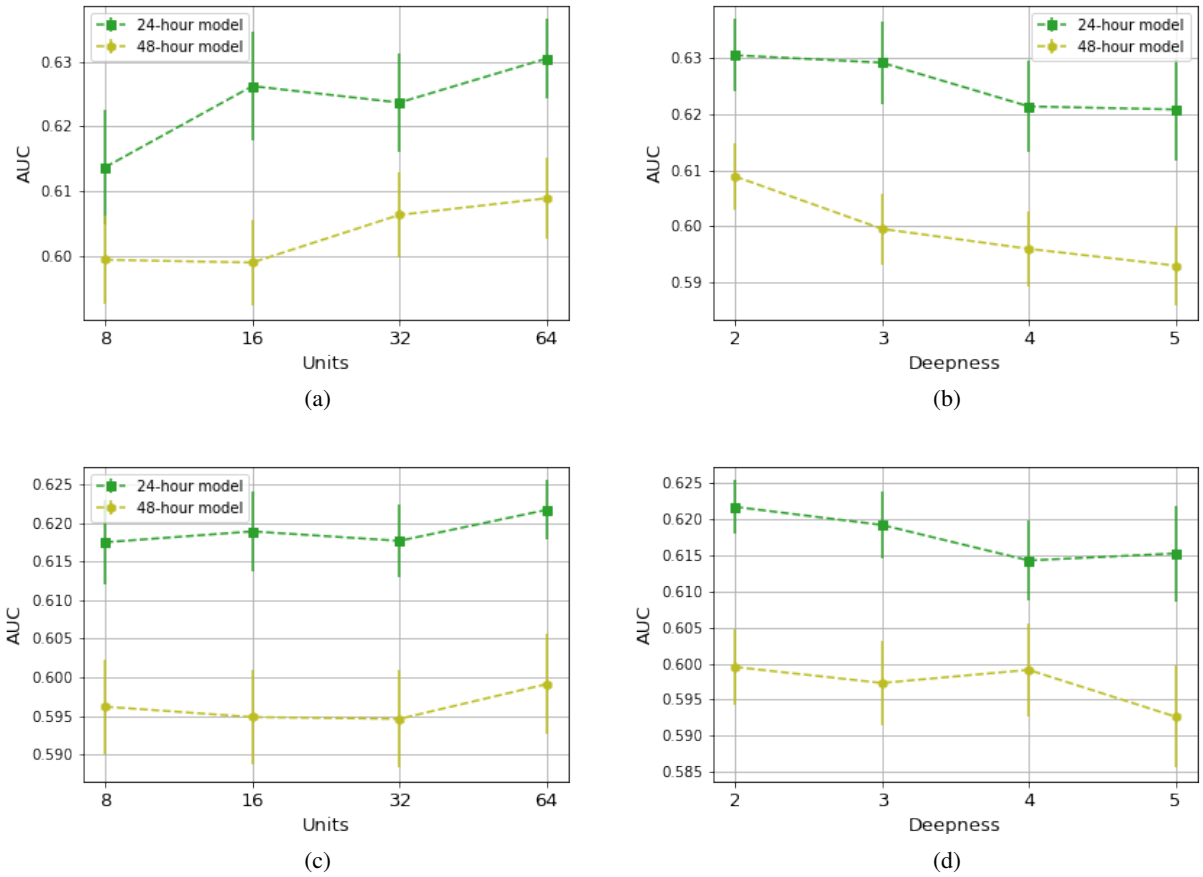


Figure 2: MLP model. Maximal AUROC as a function of the hyper-parameters *Units* (a.k.a, number of units) and *Deepness*. Each plot presents the behavior of both the 24-hour (green line) and 48-hour models (yellow line). The following cases are considered: (a) Number of units and ReLU activation function, (b) Deepness and ReLU activation function, (c) Number of units and tanh activation function, (d) Deepness and tanh activation function.

Unlike standardization (i.e., one imposes that all time-series features have unitary variance and zero mean value), the application of these data-based linear transformations does not drastically distort proper characteristics of the vital signals such as scale (i.e., the mean) and energy (i.e., the empirical second moment) value. In addition, data were processed using the Piecewise Approximate Aggregation (PPA) method (Chen and Qi, 2019). Instead of representing all very-high-scale details, we obtained a reduced but informative representation of vital signals while maintaining the lower bound of distance measurements in Euclidean space. Therefore, we used PPA to aggregate time intervals of length 9 minutes.

In Figure 3a, we see that the accuracy of the CNN increases with the number of filters: a high number of filters, such as 128, makes both 24-hour and 48-hour models accurate with AUROC 0.72 and 0.67, respectively. The composition of many hidden layers is another key feature of enabling the model to be performative. In Figure 3b we see that few layers are enough for the 24-hour model (AUROC in the range [0.70, 0.72]), whereas 6 convolutional layers are needed to enable the 48-hour model to achieve the highest AUROC (0.67). Conversely, the amplitude of the convolutional masks reduces the AUROC values, especially if one considers masks of size 17 or 33; see Figure 3c. Convolutional masks of size 3 enable keeping the AUROC 0.72 and 0.67 for both the 24-hour and 48-hour models. Thus, after searching for the best configuration, our investigation revealed that powerful (i.e., with many filters) and deep networks with small-sized kernels are the type of CNN models to use.

For completeness, we compared a pure convolutional approach (i.e., CNN model) with a CNN-LSTM model. As mentioned above, the latter has precisely the same architecture as the CNN model, except for the fact that an LSTM layer replaces the *flattern layer* of the CNN model. The LSTM layer possesses only one relevant hyper-parameter, i.e., the *number of units* denoting the number of items used to encode the impute data. Thus, we considered a CNN-LSTM

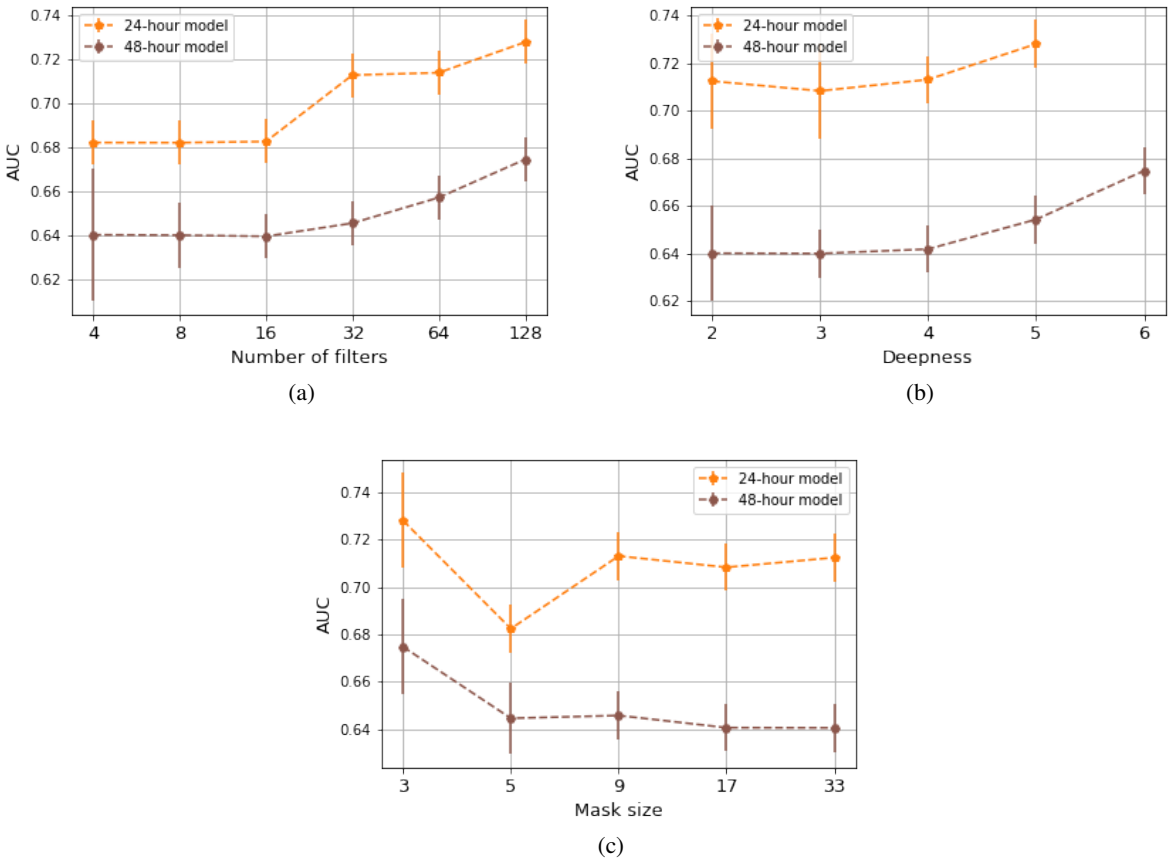


Figure 3: CNN model. Maximal AUROC as a function of the hyper-parameters *number of filters* (a), *deepness* (b), and *kernel size*. Each plot presents the behavior of both the 24-hour and 48-hour models.

with the best hyper-parameters found for the optimization of the CNN model. Still, the parameter *number of units* is the unique variable assuming different values. In Figure 4, we see that an increase of the number of units does not translate into an increase of the AUROC score. For the 24-hour model, the plateau region starting at unit 64 reveals that the CNN-LSTM model is as accurate as the CNN model, i.e., AUROC score equal to 0.72 ± 0.01. The 48-hour model cannot achieve AUROC values larger than 0.6, given any configuration of the LSTM units.

The last class of models that we tested is the *two-dimensional* CNN. Although one dimensional convolutional layers represent the most natural choice, a two dimensional convolutional-based approach is always possible if one provides a 2-D representation of the sequential data. For example, the method developed by Ye et al. (2019) offers the possibility of giving a 2-D representation of time series data, i.e. a 2-D binning is performed, where each 2-D bin counts the number of records falling in a specific range of values and at some precise moments along the time domain; see the example in Figure 5. A 2-D grid-structured representation of the EHR enabled us to investigate the possibility of using a 2-D CNN model to early identify the onset of ICU-AI.

Despite sharing a similar structure with the one-dimensional CNN, the implementation of the 2-D CNN required tuning a larger number of hyper-parameters. Such an increase in hyper-parameters is mainly due to the 2-D structure of data; unlike the 1-D case, in the 2-D case, both the width and the height of the convolutional masks need to be optimized as well as the height and the width of the 2-D bins representing each time-series feature. As with the 1-D CNN model, we optimally tuned the model on a fine grid of parameters: *number of filters, kernel size (on both the 2 dimensions), deepness, dropout, learning rate, batch size*, and both *width and height of the 2-D bins*. In Figure 6, one can see that drastic changes in the architecture of the 2-D CNN model do not cause relevant changes in the evaluation of the maximal AUROC score. That is, opting for several configurations in the number of filters (see figure 6a), in the deepness (see Figure 6b), in the size of the convolutional masks (Figure 6c), and in the height and width of the 2-D bins (Figure 6d) do not lead both the 24-hour and the 48-hour models to achieve AUROC scores larger than 0.63.

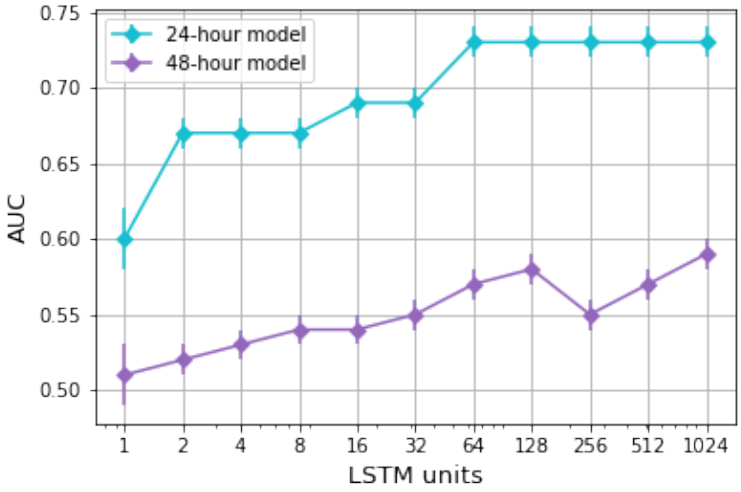


Figure 4: CNN LSTM model. AUROC as a function of the number of units of the LSTM layer. The choice of all other hyperparameters of the CNN-LSTM model is identical to the ones of the optimal CNN model.

2.1 Chosen design

Among all the models considered, the (one-dimensional) CNN model was not the one with the highest predictive performance. Although the 24-hour CNN-LSTM model could be slightly more accurate than the CNN, we observed that the latter showed more precise predictive performances even with 48-hour Time-Series instances. The difference in terms of AUROC between both models is marginal for the 24-hour model but instead evident for the 48-hours models. Moreover, we opted for a CNN model also because we want to explain the activity of pattern recognition via a robust XAI method such as the SMOE scale (see Section 3).

For the 24-hour model, we propose the following optimal architecture:

1. *Convolutional Layers*: the number of filters on each layer is 128, and each filter has a size of 3 (pixels). The result of these convolutions is referred to as *feature maps*.
2. *Activation Layer*: the ReLU function is applied after each convolution operator. This application of a non-linear activation function on the feature maps gives birth to the *activated feature maps*.
3. *Max-pooling layer*: the activated feature maps are resampled via a Max-pooling operator with a pooling size of 2.

This sequence of hidden layers is repeated five times. The architecture also encloses a *Dropout layer* after each Max-Pooling layer. The Dropout layer has a dropout rate of 0.25. The last feature map is flattened into an array and then propagated in a *fully-connected* layer (*dense layer*) with a sigmoid activation function. The activation function returns a positive output between 0 and 1, that is, the risk score denoting the chance of a patient developing an ICU-AI episode. As usual, the loss function is the binary cross-entropy, and the optimizer is the ADAM algorithm. For the 48-hour model, the architecture is identical to the 24-hour one, except for the fact that the sequence of convolutional and max-poling layers is repeated 6 times.

3 Saliency Map Order Equivalent (SMOE) scale

In this section we present the algorithm used in the manuscript for estimating the saliency maps. Differently from the gradient based methods (e.g., the *Vanilla Gradient* (VG), see for instance Simonyan et al. (2013)), the SMOE scale (Mundhenk et al., 2019) provides a different perspective for the estimation of the saliency of the CNN-activated feature maps. In fact, the SMOE scale focuses on the statistics of the activation of these feature maps.

The algorithm provides a (reasonably) faithful representation of the information contained in the input data: the larger is the overall activation of the feature maps, the more the input features are likely to be informative. Let us consider a CNN model, we denote with $\chi_{ij} \in \mathbb{R}^+$ the values of an activated feature map with ReLU activation function and

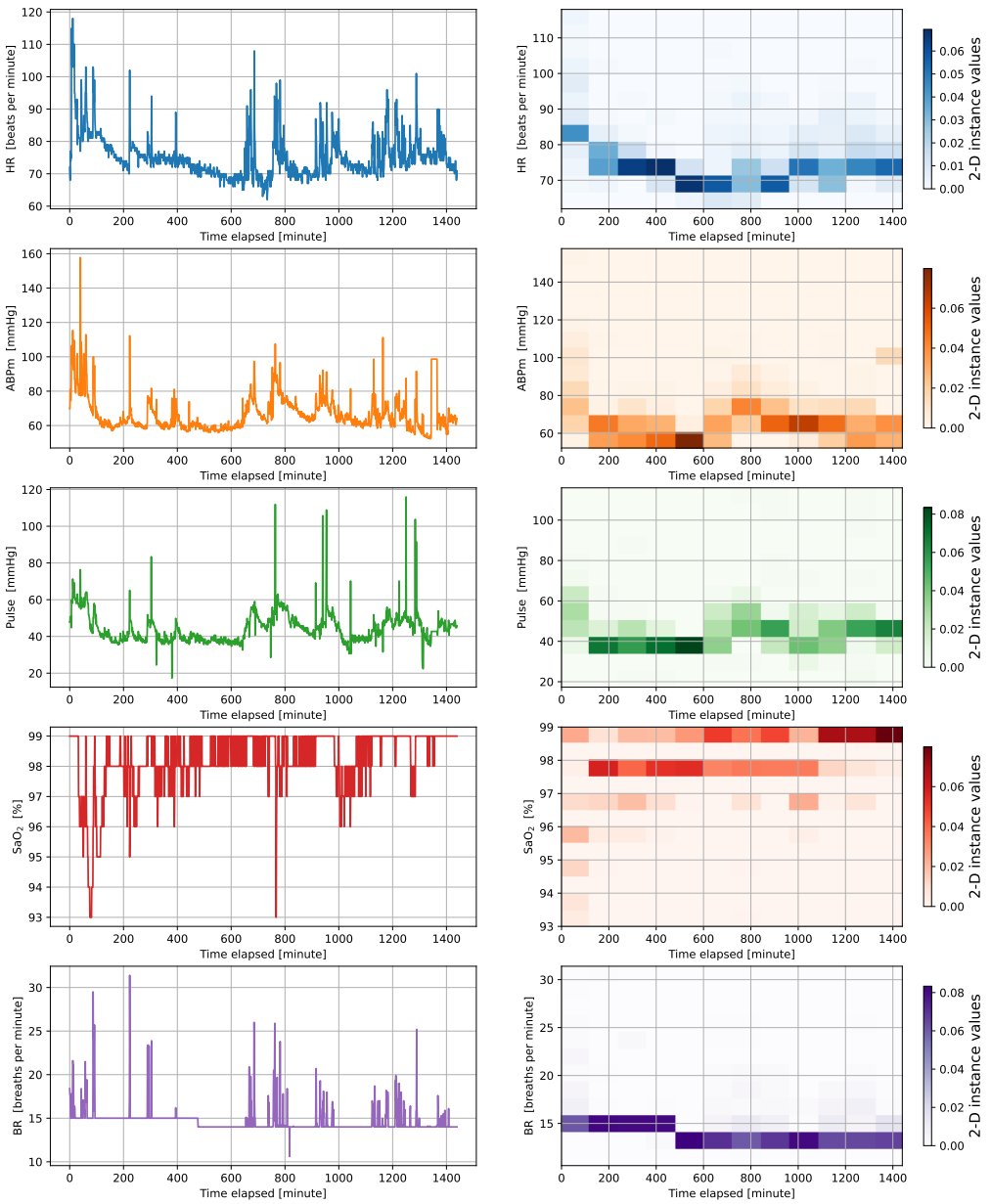


Figure 5: Example of 2-D representation of a time-series instance. On the left column the time-series features (EHR), while on the right columns their 2-D representation

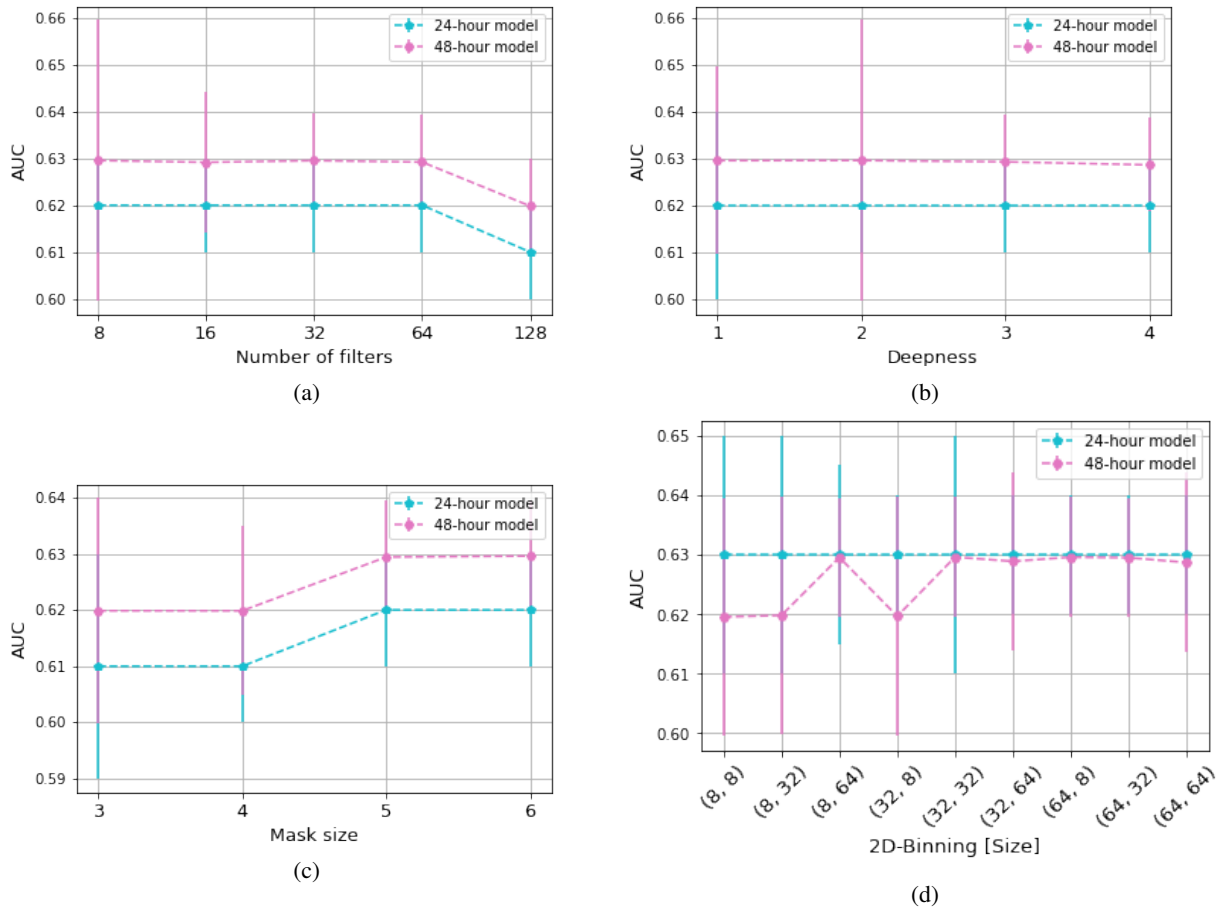


Figure 6: 2-D CNN model. Maximal AUROC as a function of the hyper-parameters *number of filters* (a), *deepness* (b), *kernel size* (c), and the *dimensionality of the 2-D bins* (d). Each plot presents the behavior of both the 24-hour (cyan) and 48-hour (pink) models.

denote with i and j, respectively, the spatial domain and the depth (i.e., number of time-series features). A function $\varphi : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is applied at each point of the spatial domain, all over the depth dimension. Thus, we obtain the saliency map via the relation $S = \varphi(\chi)$. We assume that the activated feature map χ is Gamma distributed with shape parameter k and scale parameter θ. The reason for this assumption relies on the fact that the Gamma distribution is the *maximum entropy probability distribution* for a random variable whose mean and entropy are fixed (Lagrange multipliers). Since in our context each activation map has both a fixed mean value (i.e., the scale of the activation map) and fixed entropy (i.e., the information captured in the feature map), the choice of a Gamma distributed feature map seems natural. Therefore, we estimate the distribution parameters by means of the Maximum Likelihood Principle, namely:

$$\hat{\theta}_i = \frac{\sum_{j=0}^{D} \chi_{ij}}{D\hat{k}_i}, \tag{1}$$

and

$$\log(\hat{k}_i) - \psi(\hat{k}_i) = \log\left(\frac{\sum_{j=0}^{D} \chi_{ij}}{D}\right) - \frac{\sum_{j=0}^{D} \log \chi_{ij}}{D};$$

with the sums running over the depth domain (i.e. the domain of the input features), with D the number of input features, and $\psi(x)$ the digamma function (Silverman et al., 1972). We recall that the digamma function is defined as:

$$\psi(x) = \frac{d \log \Gamma(x)}{dx},$$

with $\Gamma(x)$ the Euler's Gamma function (Silverman et al., 1972).

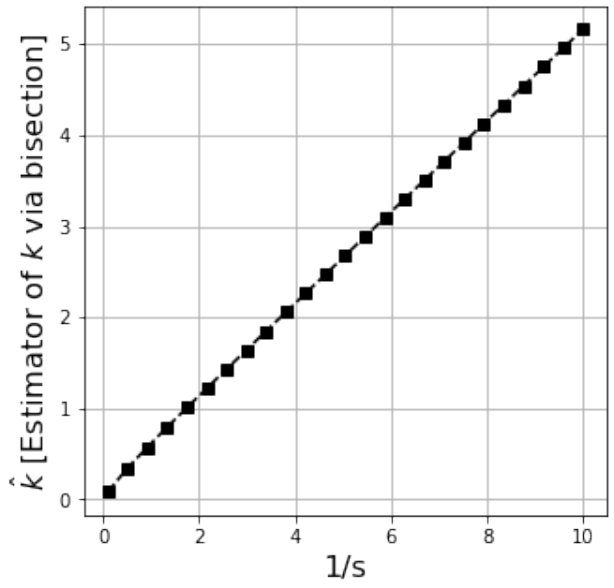


Figure 7: Estimated $\hat{k}$ via Bisection method as a function of the value $1/s$

Note that the estimation of the parameters θ_i and k_i is restricted to the i-th element along the spatial domain of the activated feature map χ: we are extracting information about the sparseness of activation along the depth domain, and not along the spatial domain. However, we cannot find an estimation of k_i in a closed form, but we can let:

$$s_i = \log\left(\frac{\sum_{j=0}^{D}\chi_{ij}}{D}\right) - \frac{\sum_{j=0}^{D}\log\chi_{ij}}{D};$$

and then we use of the asymptotic expansion of the digamma function (Abramowitz and Stegun, 1964), and we obtain the following approximation:

$$\log x - \psi(x) \simeq \frac{1}{2x}\left(1 + \frac{1}{6x}\right).$$

Thus, a first order approximation of $\hat{k}$ is given by:

$$\hat{k}_i \simeq \frac{\frac{1}{4} + \frac{1}{2}\sqrt{1+3s_i}}{s_i}. \tag{2}$$

However, we can refine $\hat{k}_i$ by using the Newton-Ralphson method (Ypma, 1995) and use (2) as an initial value. As a result, $\hat{k}_i$ appears to be related to $\frac{1}{s_i}$; as shown in Figure 7. After substituting $\frac{1}{s_i}$ with $\hat{k}_i$ in (1) we obtain:

$$\hat{\theta}_i = \left(\frac{\sum_j^D \chi_{ij}}{D}\right)\left[\log\left(\frac{\sum_j^D \chi_{ij}}{D}\right) - \sum_{j=0}^{D}\frac{\log\chi_{ij}}{D}\right],$$

which can be finally rewritten as:

$$\hat{\theta}_{\text{SMOE},i} = \frac{1}{D}\sum_{j=0}^{D}\langle\chi\rangle\log\frac{\langle\chi\rangle}{\chi_{ij}}, \tag{3}$$

where

$$\langle\chi\rangle := \frac{1}{D}\sum_{m=0}^{D}\chi_{lm}.$$

Hence, (3) represents the *SMOE scale*, i.e. the statistics involved in the computation of the saliency maps: it is proportional to the activated mean value (along depth) via the term $\langle\chi\rangle$. Moreover, it depends on the the variance, as we can see by performing a Taylor expansion of $\log\chi_{ij}$ around $\langle\chi\rangle$:

$$\log\langle\chi\rangle - \langle\log\chi\rangle \simeq \frac{\langle\chi - \langle\chi\rangle\rangle^2}{2\langle\chi^2\rangle}.$$

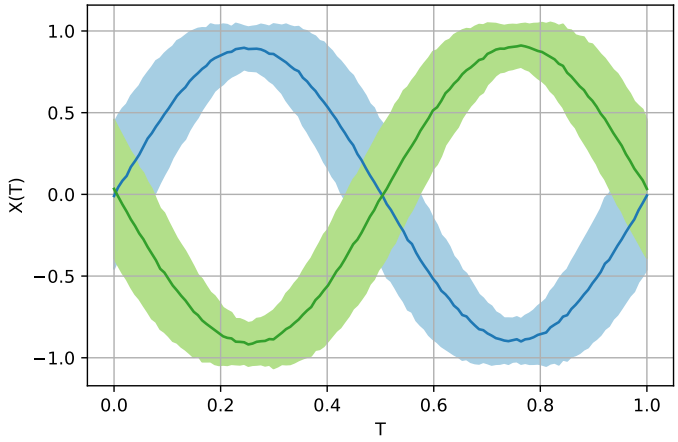


Figure 8: Example of the toy sine dataset. The mean samples are plotted in blue and green, respectively, for classes 0 and 1. The light-colored areas represent the standard deviation values of the mean samples.

The simplification is used in the estimation of the Gamma scale parameters is the SMOE to the full iterative scale parameter estimation (Mundhenk et al., 2019).

By construction, we can apply the SMOE scale only on one single activated feature map; that is, we can only estimate the informative sparseness of each activated feature map independently. We can then combine them in order to obtain an overall measurement of saliency at each spatial/temporal location.

Therefore, the SMOE scale used is the estimated scale parameter of a Gamma distribution. In Section 6 of the main manuscript we have used this assumption for deriving the saliency map. However, we have checked this hypothesis via a multiple Kolmogorov-Smirnov test, with Bonferroni correction: for each activation map and for a given value of the temporal domain, we have tested whether the values of the feature are gamma distributed with scale parameter $\hat{\theta}_i$ as estimated in (3). We did not reject the null hypothesis with $\alpha = 0.05$.

Before concluding this section, we want to present a brief example of the XAI methods we have introduced above. Let us consider the following toy data set for binary classification of time-series: the class 0 is generated by:

$$X_0^{(n)}(t) = \sin\left(2\pi[t + \phi_n]\right),\ t \in [0, 1];$$

with $X_0^{(n)}(t)$ denoting the n-th instance of the class 0, and $\phi_n \overset{i.i.d}{\sim} U(-0.125, 0.125)$. Likewise, for class 1 we set:

$$X_1^{(n)}(t) = -\sin\left(2\pi[t + \phi_n]\right),\ t \in [0, 1].$$

We shall refer to this dataset as *toy sine dataset*. A representation of the toy sine dataset is shown in Figure 8. We train and test a CNN with a one-held-out approach (i.e., we only evaluate the model's accuracy after making just one split into train and test set) with train size 75% (i.e., we use 75% of the dataset to train the CNN model). As a result, the AUROC of the model is equal to 0.99.

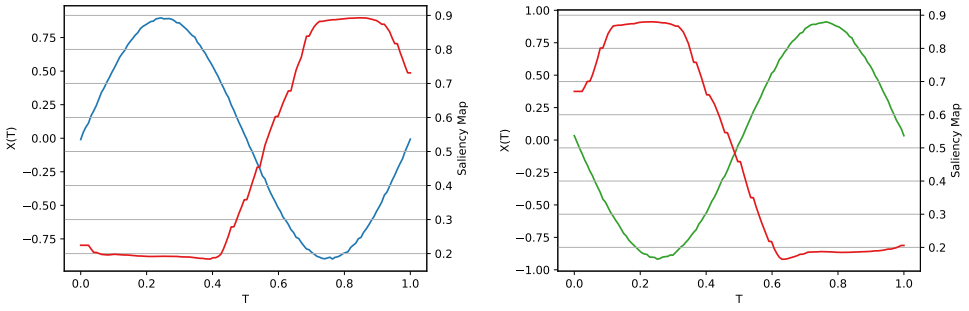
Figure 9 shows the saliency maps for the SMOE Scale. We notice that that the saliency maps detect a salient area in correspondence with the trough of the sinusoidal oscillation, i.e., those areas of the input domain where the saliency maps achieve the highest values. As expected, the localization of the trough in two distinct areas of the input domain $T \in [0, 1]$ represents the critical feature that the CNN captures to distinguish the two classes.

References

Abramowitz, M. and Stegun, I. A. (1964). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, volume 55. US Government printing office.

Chen, Y. and Qi, B. (2019). Representation learning in intraoperative vital signs for heart failure risk prediction. *BMC medical informatics and decision making*, 19(1):1–15.

Mundhenk, T. N., Chen, B. Y., and Friedland, G. (2019). Efficient saliency maps for explainable ai. *arXiv preprint arXiv:1911.11293*.



(a) (b)

Figure 9: Saliency maps obtained via SMOE Scale method for the sine toy dataset. (a) Mean sample (blue) and mean saliency map (red) for class 0. (b) Mean sample (blue) and mean saliency map (red) for class 1.

Silverman, R. A. et al. (1972). *Special functions and their applications*. Courier Corporation.

Simonyan, K., Vedaldi, A., and Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*.

Ye, Y., Jiang, J., Ge, B., Dou, Y., and Yang, K. (2019). Similarity measures for time series data classification using grid representation and matrix distance. *Knowledge and Information Systems*, 60(2):1105–1134.

Ypma, T. J. (1995). Historical development of the newton–raphson method. *SIAM review*, 37(4):531–551.