Ulf Brefeld · Edward Curry
Elizabeth Daly · Brian MacNamee
Alice Marascu · Fabio Pinelli
Michele Berlingerio · Neil Hurley (Eds.)

LNAI 11053

Machine Learning and Knowledge Discovery in Databases

European Conference, ECML PKDD 2018
Dublin, Ireland, September 10–14, 2018
Proceedings, Part III

3 Part III

Springer

Hypotensive Episode Prediction in ICUs via Observation Window Splitting

Elad Tsur[1], Mark Last[1](✉), Victor F. Garcia[2], Raphael Udassin[3], Moti Klein[4], and Evgeni Brotfain[4]

[1] Department of Software and Information Systems Engineering, Ben-Gurion University of the Negev, 84105 Beer-Sheva, Israel
eladtsur@gmail.com, mlast@bgu.ac.il

[2] Division of Pediatric Surgery, MLC 2023, Children's Hospital Medical Center, 3333 Burnet Avenue, Cincinnati, OH 45229, USA
victor.garcia@cchmc.org

[3] Pediatric Surgery Department, Hadassah University Hospital, Ein-Karem, 9112001 Jerusalem, Israel
raphaelu@ekmd.huji.ac.il

[4] General Intensive Care Unit, Soroka Medical Center, Beer Sheva, Israel
{MotiK,EvgeniBr}@clalit.org.il

Abstract. Hypotension, defined as dangerously low blood pressure, is a significant risk factor in intensive care units (ICUs), which requires a prompt therapeutic intervention. The goal of our research is to predict an impending Hypotensive Episode (HE) by time series analysis of continuously monitored physiological vital signs. Our prognostic model is based on the last Observation Window (OW) at the prediction time. Existing clinical episode prediction studies used a single OW of 5–120 min to extract predictive features, with no significant improvement reported when longer OWs were used. In this work we have developed the *In-Window Segmentation* (InWiSe) method for time series prediction, which splits a single OW into several sub-windows of equal size. The resulting feature set combines the features extracted from each observation sub-window and then this combined set is used by the Extreme Gradient Boosting (XGBoost) binary classifier to produce an episode prediction model. We evaluate the proposed approach on three retrospective ICU datasets (extracted from MIMIC II, Soroka and Hadassah databases) using cross-validation on each dataset separately, as well as by cross-dataset validation. The results show that InWiSe is superior to existing methods in terms of the area under the ROC curve (AUC).

Keywords: Time series analysis · Clinical episode prediction
Feature extraction · Intensive care · Patient monitoring

Partially supported by the Cincinnati Children's Hospital Medical Center; In collaboration with Soroka Medical Center in Beer-Sheva and Hadassah University Hospital, Ein Karem, Jerusalem

U. Brefeld et al. (Eds.): ECML PKDD 2018, LNAI 11053, pp. 472–487, 2019.
https://doi.org/10.1007/978-3-030-10997-4_29

1 Introduction

Hypotension is defined as dangerously low blood pressure. It is a major hemodynamic instability symptom, as well as a significant risk factor in hospital mortality at intensive care units (ICUs) [1]. As a critical condition, which may result in a fatal deterioration, an impending Hypotensive Episode (HE) requires a prompt therapeutic intervention [2] by ICU clinicians. However, HE prediction is a challenging task [3]. While the clinical staff time is limited, the amount of accumulated physiologic data per patient is massive in terms of both data variety (multi-channel waveforms, laboratory results, medication records, nursing notes, etc.) and data volume (length of waveform time series). Even with sufficient time, resources, and data, it is very hard to accurately estimate the likelihood of clinical deterioration with bare-eye analysis alone.

HE may be detectable in advance by automatic analysis of continuously monitored physiologic data; more specifically, the analysis of vital signs (multi-parameter temporal vital data), may inform on the underlying dynamics of organs and cardiovascular system functioning. Particularly, vital signs may contain subtle patterns which point to an impending instability [4]. Such pattern identification is a suitable task for machine learning algorithms. Smart patient monitoring software that could predict the clinical deterioration of high risk patients well before there are changes in the parameters displayed by the current ICU monitors would save lives, reduce hospitalization costs, and contribute to better patient outcomes [5].

Our research goal is to give the physicians an early warning of an impending HE by building a prediction model, which utilizes the maximal amount of information from the currently available patient monitoring data and outperforms state-of-the-art HE prediction systems. We present and evaluate the *In-Window Segmentation* (InWiSe) algorithm for HE prediction, which extracts predictive features from a set of multiple observation sub-windows rather than from a single long observation window.

This paper is organized as follows. Section 2 surveys the previous works in several related areas, elaborates on the limitations of these works and introduces the contributions of our method. Section 3 describes the studied problem and proposed methods in detail and Sect. 4 covers the results of an empirical evaluation. Finally, Sect. 5 presents the conclusions along with possible directions for future research.

2 Related Work and Original Contributions

Several works studied the problem of clinical deterioration prediction in ICUs. This section reviews their problem definitions, feature extraction methods, sliding window constellations, and prediction methodologies. Finally, a discussion of the limitations of existing methods is followed by a presentation of the contributions of this study.

2.1 Clinical Episode Definitions

Previous works on clinical deterioration prediction vary mainly in two aspects [6]. The first one is an episode definition, which may be based on the recorded clinical treatment or on the behavior of vital signs within a specific time interval. The second one is the warning time, a.k.a. the *Gap Window*, which will be called in brief the *gap* in this study.

The objective in [3] was to predict the hemodynamic instability start time with a 2-h gap. The episode start time was defined by a clinical intervention recorded in the ICU clinical record of a patient. In [7], instability was also defined by some given medications and gaps of 15 min to 12 h were explored.

The 10th annual PhysioNet/Computers in Cardiology Challenge [4] conducted a competition to study an Acute Hypotensive Episode (AHE). They defined AHE as an interval, in which at least 90% of the time the Mean Arterial blood Pressure (MAP) is under 60 mmHg during any 30-min window within the interval. Their goal was to predict whether an AHE will start in the next 60 min. In [1,8], the HE and AHE definitions were identical to [4], but a lower MAP bound of 10 mmHg was added to prevent noise effects from outliers. Their goal was to predict the patient condition in a *Target Window* (called herein *target*) of 30 min, which occurs within a gap of 1–2 h (See Fig. 1a), and label it as hypotensive or normotensive (normal blood pressure). As expected, and as concluded in [8], the problem is more challenging when predicting further into the future, thus resulting in poorer performance. Note that, as indicated in [8], the accepted HE definitions for adults vary in the range of 60–80 mmHg MAP for 30+ min, where the lowest case of 60 mmHg is sometimes excluded under the definition of AHE [1,4].

2.2 Predictive Feature Types

In most works, the future episode predictive features are usually extracted from a sliding *Observation Window* (OW) over a record, which is a collection of vital sign time series of one patient in a single ICU admission. A minute-by-minute vital signs time series, like blood pressure and Heart Rate (HR) are usually used to extract features, while a few studies used the clinical information (age and temporal medications data) as well. A typically used benchmark database is MIMIC II [9], a multi-parameter ICU waveforms and clinical database.

Statistical features are the most obvious source for the extraction of predictive features, also called patterns, from intervals like OWs. In [5], the authors calculate extremes, moments, percentiles and inter-percentile ranges for every vital sign, whereas in [8] interquartile ranges and slope are extracted as well. In a more pragmatic statistical approach [10], several episode predictive indices were used, derived from the blood pressure signals only. These indices were six types of averages from Systolic Blood Pressure (SBP), Diastolic Blood Pressure (DBP), and MAP, each taken as a single feature. Another statistical approach derives *cross-correlation* features which capture the coupling between two time

series by computing the sum of products of their values [8], or by estimating their variance and covariance [5].

A more recent and widely accepted feature extraction approach is the use of *wavelets*, which captures the relative energies in different spectral bands that are localized in both time and frequency. Wavelets were proven to perform well as episode predictors [11] as well as vital sign similarity detectors [12]. In [5] and [8], Daubechies (DB) and Meyer wavelet types were used, respectively, noting that the DB type dominates the basic Haar type wavelets [13] in terms of vital sign time series, which are non-stationary [5].

Apart from vital signs, patient age and vasopressors (blood pressure medications) given during OWs are added as features by Lee and Mark [8] but found to have low correlation with the target. In their other work [15], they achieve similar results without those features. Moreover, Saeed [11] mentions the low reliability of vasopressor medication timestamps, which are very important for the episode prediction task.

2.3 Observation Window Constellations

The sliding OW plays an important role in the episode prediction task. In this section, we survey the different approaches to constructing and collecting OWs.

The first important attribute of an OW is its duration. In [1,3,5,8,10,14], various OW sizes were applied (5, 10, 30, 60, 90, and 120 min). Having implemented a 60-min OW, it is claimed in [1,8] that extracting features from a longer window does not result in improvement of prediction performance.

In [15], Lee and Mark extended their previous work by implementing a weighted decision classifier that consists of four base classifiers, each predicting an HE in the same target but within different gaps (1, 2, 3 and 4 h) using a different corresponding 30-min OW. The final decision is made by weighting the four posterior probabilities from each classifier. They report insignificant improvement in prediction performance as well as independency of predictions from the 3rd and the 4th past hours (the earliest hours).

A second matter is how to collect OWs for training the prediction model. One simple approach, applied by Cao et al. [3], is to compile an OW ending gap-minutes before the start time of the first episode in every unstable record (having one or more HEs). For each stable record, one or more OWs are then sampled randomly. According to [8], collecting multiple OWs from both stable and unstable records in a random fashion, which does not collect windows exactly gap-minutes before an episode start, is proved to outperform the first method of Cao [3]. A sliding target window (with no overlap) traverses each record (Fig. 1a), and as many OWs as possible are compiled. However, they note that collecting OWs all the time, even when no HE is impending, and doing it from both stable and unstable patients will result in an extremely imbalanced dataset. Having two OW classes (hypotensive or normotensive), one way to solve this issue is by undersampling the majority class (normotensive) [5,8].

2.4 Prediction Methodology

The problem of episode prediction is most logically approached by calculating the probability of a future episode at a given time point, using features extracted from the current OW, and then classifying the target window, which starts within some gap-minutes, as hypotensive or not, based on a pre-defined probability threshold. Multiple works tackled this problem by using numerous supervised machine learning algorithms, particularly binary classifiers, with some exceptions such as in [16], where a Recurrent Neural Networks approach is used to forecast the target window MAP values, followed by a straightforward binary decision based on the episode definition.

The classifiers used by some other papers are Logistic Regression [3], Artificial Neural Network [8,15], Majority Vote [1], and Random Forest [5]. To the best of our knowledge, the most accurate HE prediction so far is reported in [8,15], which we reproduce and use for comparison in Sect. 4.

2.5 Limitations of Published Methods

Advanced methods and evaluation schemes such as in [1,5,8,14,15], solved some of the problems found in the early works [3,10], yet left some open issues, including low precision (14% in [8]) and a strict episode definition that is still far from the practical definitions used in ICUs. Moreover, a machine learning solution to a high precision HE prediction will probably need much more training data, while the current MIMIC II [9] contains only several thousands[1] of vital sign records that are matched with the clinical data.

Unfortunately, there is a lack of comprehensive public sources of ICU monitored vital signs beyond the existing MIMIC database. Consequently, the current episode prediction studies miss the crucial cross-dataset validation, which is needed to find a generic model that should work for any ICU, disregarding availability of retrospective patient records for training.

Recent papers [1,5,8] include predictions of future episodes even if the patient is going through an ongoing episode. These predictions may be less relevant to the physicians and possibly excluded from the evaluation metrics.

Finally, studies conducted over the last decade show no improvement in utilizing OWs greater than 120 min (and usually even 60 min), implying there are no additional predictive patterns to be found in the near past. On the contrary, the results from [1,5,7,8,14] show an accuracy decrease of only 1–2.5% when switching from a 60-min gap window to a 120-min one, which may imply that earlier observations may have a just a slightly lower correlation to the target. Thus, there may be additional predictive patterns, which are not utilized properly by the existing methods.

[1] Recently, The MIMIC III waveform database *Matched Subset*, four times larger than the MIMIC II subset, was published

2.6 Original Contributions

The main contribution of this paper is the In-Window Segmentation (*InWiSe*) method, which aims to utilize the local predictive patterns in long OWs. The method, presented in Sect. 3.2 and Fig. 1b, differs from previous methods by the following: (i) it extracts hidden local features by splitting the OW into multiple sub-windows, which improves the model predictive performance; (ii) it is flexible in terms of OW definition - if a complete sub-window set is not valid for use at the prediction time, a single OW option is used instead.

As mentioned in Sect. 2.3, a step towards multiple OW utilization was taken in [15] by combining weighted predicted posteriors of four OWs, each making an independent prediction with a distinct gap. Their approach is different from ours mainly in that we let the classifier learn the association between cross-window features, which is not possible in a weighted posterior decision. Another very recent work (DC-Prophet) [19], published while writing this paper, combines features from consecutive time series intervals (lags) to make early predictions of server failures. Their approach is similar to ours, but it has neither been applied to clinical episode prediction, nor it has handled invalid lags.

A further contribution of our work is evaluation of both our method and earlier episode prediction methods in a *cross-dataset* setting, in addition to the in-dataset cross-validation. Finally, our experiments are extended by a new evaluation approach, which excludes the clinically irrelevant in-episode predictions.

3 Methodology

This section starts with the problem definition, continues with introducing InWiSe, and concludes with the description of the data compilation, feature extraction, and classification methods used in this study.

3.1 Problem Definition and Prediction Modes

This study explores the problem of predicting a patient condition (hypotensive or normotensive) within a 60-min gap. Following the work in [1,8], we define a hypotensive episode (HE) as a 30-min target window where at least 90% of MAP values are below 60 mmHg. Any valid target (see the validity definition in Sect. 4.2) not meeting this criterion is labeled as normotensive. At the prediction time, each sub-window set is labeled with respect to its corresponding target.

Considering the implementation of the proposed method in the clinical setting, we distinguish between two alternative prediction modes: (i) *all-time prediction*, where the assumption (found in previous papers) is that episode prediction is needed continuously regardless of the clinical condition at the prediction time; (ii) *exclusive prediction*, where episode prediction is needed only when the patient is not in a currently recognized HE (the last 30 min of the ICU stay are not an HE by definition).

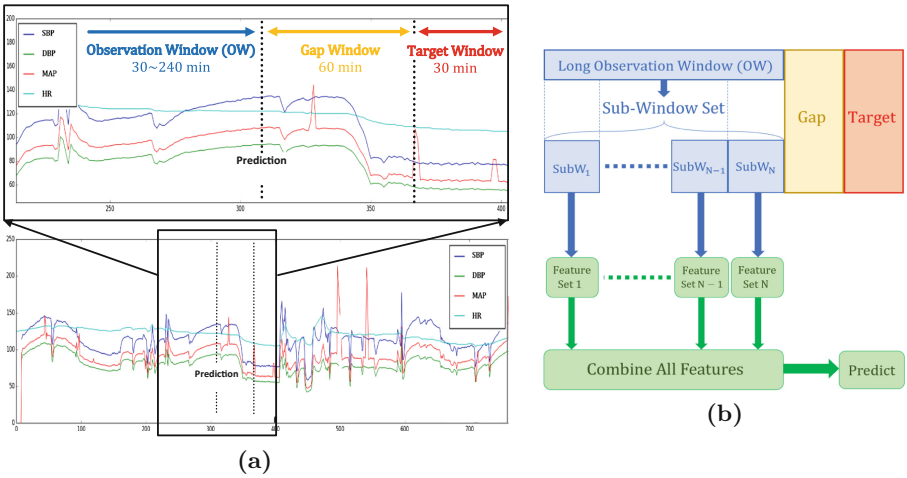

Fig. 1. (a) Basic Method: traversing over a patient record with an impending HE is demonstrated by the observation, gap and target windows with respect to the prediction time. (b) InWiSe: a given OW is split into a sub-window set of size N, followed by a prediction that is based on the combined feature set of the N sub-windows ($SubW$s).

3.2 Splitting Obervation Windows with InWiSe

In our study, which was developed based on the observation-gap-target windows scheme demonstrated in Fig. 1a, we hypothesized that taking longer OWs, splitting them into several equally sized sub-windows, also called the *sub-window set*, and combining all their features together (see Fig. 1b) would improve the predictive accuracy of the induced model versus using a smaller feature set of a single long or short OW. For example, a set of the mean MAP values from four, three, two and one hours before the same target window may be more informative for predicting the target label than the mean MAP value of a single 4-h OW.

The InWiSe method does not use a classifier based on a combined set of features if one of the current sub-windows in the set is invalid (see Sect. 4.2). In that case, the prediction is made by a simpler classification model using only the features extracted from the latest sub-window ($SubW_N$ in Fig. 1b) unless that window is invalid. Consequently, InWiSe misses less prediction points than the single OW method (more details are in Sect. 4.4, in-dataset paragraph).

3.3 Feature Extraction

We use three basic vital signs (SBP, DBP, and HR) to derive two additional vital signs for each record: Pulse Pressure calculated by $PP = SBP - DBP$, and Relative Cardiac Output calculated by $CO = HR \times PP$. Next, we extract the following three groups of features from each sub-window.

Statistical Features: mean, median, standard deviation (Std), variance, interquartile range (Iqr), skewness, kurtosis and linear regression slope are calculated for each of the vital signs. Missing values are ignored.

Wavelet Features: Similarly to [5], multi-level discrete decomposition of each vital sign can be conducted with DB wavelets. The decomposition of a single time series (signal) X is denoted by $W_X = [a_n\ d_n\ d_{n-1}\ \ldots\ d_1]$, where n is the decomposition level (window size depended), a_n is the signal approximation, and d_k is the detail signal of level k. The elements in W_X are then utilized as features by calculating the relative energy for each of them as in [8,15]. Missing values are interpolated.

Cross-Correlation Features: the cross correlation of two time series $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ is defined as $\rho_{XY} = \frac{1}{n}\Sigma x_i y_i$ and calculated for each pair of vital signs.

The total amount of features extracted from a sub-window set is equal to the number of sub-windows N multiplied by the feature set size.

3.4 Classification

Each instance in the training dataset is composed of a sub-window set feature vector and a class label which is positive or negative (the target is either hypotensive or normotensive, respectively). Before training a binary classifier, we both normalize the training dataset (to zero mean and unit standard deviation) and undersample it to overcome the imbalance issue (Sect. 4.3).

Our classifier produces a posterior probability of the positive class, which may lead to an HE alert depending on the probability threshold determined from the Receiver Operating Characteristic (ROC) curve built on the target (testing) dataset. Following [3,8,15], the following selection criterion for the optimal threshold can be used: $Th_{selected} = \text{argmax}_{Th}\{sensitivity(Th) + specificity(Th)\}$.

4 Experiments

The experimental setup of this study includes multiple prediction modes, methods, and model configurations. We first perform an in-dataset evaluation for each prediction mode (all-time and exclusive) and for each method (single OW and InWiSe). Next, we proceed with a cross-dataset validation for each dataset pair. This section describes the datasets and their compiled OW and window-set statistics, followed by the experiments and analysis of results.

4.1 Data Description

Three databases of adult ICU admission records were prepared for this study: *Soroka* Medical Center in Beer Sheva (4,757 records), *Hadassah* Hospital, Ein-Karem, Jerusalem (8,366 records), and *MIMIC II* [9] (downloaded from [17,

18] and comprising 5,266 records). All time-series sampling rates are minute-by-minute (some second-by-second MIMIC II records were undersampled by taking each minute median). The common-shared vital signs among the three databases are HR, SBP, DBP, MAP, peripheral capillary oxygen saturation and respiration. Similarly to Lee and Mark [8,15], we included only the HR, SBP, DBP and MAP vital signs in our data.

4.2 Data Compilation

As a pre-processing step, any outlier (out of the rage 10–200 for any vital sign) is considered as a 'missing value'. When compiling OWs from each admission record we used the observation-gap-target window scheme (Sect. 3.2) called the *single OW method*, as well as a first step for InWiSe. The window sizes of the single OW were 30, 60, 120 or 240 min, while the gap and target sizes were constant at 60 and 30 min, respectively. Furthermore, we followed Lee and Mark [8] who claimed that a prediction rate of every 30 min should represent the performance of a real-time system. Therefore, a 30-min. sliding target window was traversed with no overlaps over each record and as many OWs as possible were compiled, depending on the prediction mode. Following [8], targets with more than 10% missing MAP values were excluded from this study, as their true labels are unknown. Turning to OW validity, to prevent a classifier from learning outlier instances, more than 5% missing values for any vital sign made the window invalid and, consequently, excluded it from our work as well.

Five InWiSe configurations of window splitting were selected for this study: 60[m] $\rightarrow$ 2 $\times$ 30[m], 120 $\rightarrow$ 2 $\times$ 60, 120 $\rightarrow$ 4 $\times$ 30, 240 $\rightarrow$ 2 $\times$ 120 and 240 $\rightarrow$ 4 $\times$ 60. For each configuration and for every record in the dataset, at the prediction time, we combine a sub-window set ($[SubW_1, \ \ldots \ , SubW_N]$, Fig. 1b) if all N sub-windows are valid. The label of a complete sub-window set is the same as of its latest sub-window, which is labeled according to its corresponding target window.

Following the window label counts, the imbalance ratio for the all-time compilation was found to be 1:20 to 1:40 in favor of normotensive (negative) windows (increasing with the observation window size), as opposed to the exclusive compilation (no in-episode predictions), which was two times more imbalanced. As for the window set method, the bigger the set size N the less positive and negative examples are available. Like in a single OW, we observed an increase in the window-set labeling imbalance with an increase in the window size that reached 1:100 in the exclusive mode.

The reduction of sub-window sets availability with increasing N varied over datasets and was caused by differences in the amount of missing values (i.e., MIMIC II misses more values than Soroka). Moreover, the reason behind cross-dataset differences in terms of total OW count with respect to record count was the variance in ICU stay duration, which was higher in Soroka than in other datasets. Last, we note that using the exclusive mode resulted in a decrease of over 50% in the positive window count, probably because the average HE duration was much longer than 30 min (i.e., Hadassah average HE duration was

98 min), increasing the time intervals where we do not make a prediction under this mode.

4.3 Experimental Setup

In-dataset Evaluation: For each dataset, mode and algorithm a 5-fold cross-validation (CV) was performed. To allow the classifier to successfully learn the imbalanced data, training folds were undersampled (5 times without replacement) to attain equal counts of stable and unstable admission records within each training fold (test folds were left unbalanced). Moreover, for each record all OWs or sub-window sets were either in training or test dataset to prevent record characteristics from leaking into the test dataset. In total, the classifier produced 25 outputs (5 folds $\times$ 5 samples) which were evaluated by five metrics: area under the ROC curve (AUC), accuracy, sensitivity, specificity and precision. Furthermore, to compare between the two methods fairly, we optimized the hyper-parameters of each classifier: an inner 5-fold CV was utilized in each undersampled training fold of the outer CV and the best chosen hyper-parameters found by a grid search were used to train the outer fold (Nested CV).

To choose the prediction model, three classifiers were evaluated using a 60-min OW (single OW method) and a 4×60-min set (InWiSe) on all datasets combined and in the all-time prediction mode (with hyper parameters optimization). The AUCs were (0.932, 0.936) for Random Forest, (0.937, 0.940) for Artificial Neural Network (ANN) with a single hidden layer of 100 neurons, and (0.939, 0.943) for Extreme Gradient Boosting (XGBoost) [20], where each tuple represents a <single OW, sub-window set> pair. Since XGBoost outperformed Random Forest and ANN with p-values = 0.03 and 0.08, respectively (using t-test), we chose XGBoost for inducing prediction models in this study. Still, ANN was used as a baseline of [8].

XGBoost is a scalable implementation of the Gradient Boosting ensemble method that affords some additional capabilities like feature sampling (in addition to instance sampling) for each tree in the ensemble, making it even more robust to feature dimensionality and helping to avoid overfitting. Moreover, considering our minimum training dataset size of approximately 2k instances together with the maximal feature vector size of 392 features, the built-in feature selection capability of XGBoost is important.

The XGBoost classifier was grid-search optimized for each dataset or mode and for each OW size or sub-window set configuration C, where the best hyper-parameters were reproduced for all datasets, in most of the CV folds. The optimized hyper-parameters were: number of trees (500, **1000**, 1500), maximum tree depth (**3**, 5), learning rate (0.001, **0.01**, 0.1), instance sample rate (0.8, **0.4**), and feature sample rate (0.9, **0.6**). The best choices are shown above in bold.

Finally, each algorithm was tried with several OW sizes and sub-window sets Cs: four OW sizes for the single OW method and five Cs for InWiSe sub-window sets (see Sect. 4.2). As a result, a total of 54 in-dataset CVs were conducted (3 datasets $\times$ 2 modes $\times$ 9 window-set Cs and single OW sizes).

Cross-Dataset Validation: The model induced from each dataset was evaluated on other datasets using the all-time mode. XGBoost was trained on one full dataset and tested on the two other datasets separately. The source dataset was undersampled only once, justified by a mostly very low variance of AUC (<0.1%) between undersamples, in each fold of the in-dataset CV. Both the single OW size and the window-set C were chosen to optimize the AUC performance in the in-dataset evaluation: 120/240-min sized OW for the single OW method and 4×60-min sub-window set for InWiSe. The hyper-parameters of the classifier (XGBoost) of each method were chosen by a majority vote over the folds in the in-dataset evaluation. A total of 18 experiments were performed (3 training datasets $\times$ 2 test datasets $\times$ 3 window-set Cs and single OW sizes).

4.4 Analysis of Results

This subsection presents the results, followed by the feature importance analysis. The reader should recall that all sub-window set results include some test instances which were classified using the latest sub-window ($SubW_N$ in Fig. 1b), if valid, in case that the sub-window set was invalid.

In-dataset: As a baseline, we reproduced the single OW method results of Lee and Mark [8] on MIMIC II with ANN. In Fig. 2, we use MIMIC II to compare the single OW method with ANN, using two more methods: single OW with XGBoost (single OW method) and sub-window set best split using XGBoost as well. In comparison with the baseline, the AUC of the 4×60 sub-window set (XGBoost) was significantly higher than for the single OW method with ANN (60, 120 or 240-min OW size) with p-values 0.009 and 0.05 for all-time and exclusive modes, respectively.

From Figs. 2 and 3, we first conclude that in terms of AUC, splitting a single OW into several sub-windows is usually better than using a single OW; we note that, the advantage of OW splitting grows with an increase in the OW duration, which emphasizes the benefit from splitting a single OW that is longer than the longest OWs used by current methods ($240 \rightarrow 4 \times 60$ versus 120 min).

Turning to XGBoost-only comparison on MIMIC II, InWiSe outperformed the single OW in the all-time prediction mode, but only with a p-value of 0.13, while performing only slightly better than the single OW in the exclusive mode (in terms of AUC). However, while these all-time prediction trends are similar in the Soroka dataset where InWiSe is better with $p = 0.15$ (Fig. 3), in the Hadassah dataset InWiSe significantly outperforms the single OW method with a p-value of 0.03. In addition, in Table 1 that shows the in-dataset best results on its diagonal, we see that, although not always statistically significant, the sub-window set method is better than the single OW alternatives in each dataset and in all metrics (in bold) when evaluating in the all-time prediction mode. Note that our significance test comparisons were between the best results of each method rather than sub-window set versus its matching long window and they were calculated with four degrees of freedom (due to five folds).

As for the exclusive mode, the AUC was lower, as expected, since less positive OW instances were available, making the prediction task harder in general.

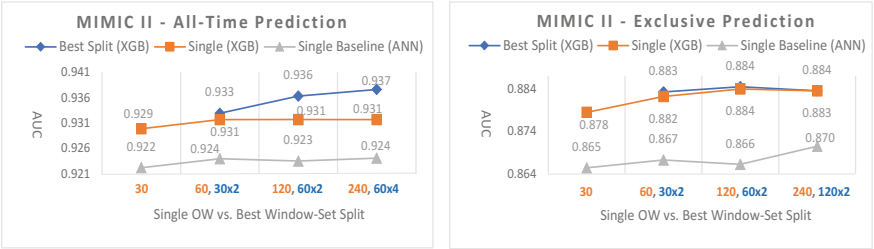

Fig. 2. The single OW method compared with its InWiSe best split, and with the ANN baseline on MIMIC II (all-time mode at the left and exclusive mode to the right).

The smaller *improvement* of InWiSe in comparison with a single OW is probably related to the considerable decrease in available positive sub-window sets relatively to the single OW count, compared to a small decrease in the case of the all-time mode. For example, Soroka HE labeled OW count decreases by 5-7% for the all-time mode, but by 35-45% for the exclusive mode, between the 4-sub-window set and the single OW methods. Table 2 shows further in-dataset metric results as Table 1 did for the all-time mode. In contrast to the all-time results, in the exclusive mode the AUC is still better with InWiSe (relatively to single OW), whereas other metrics domination depends on the posterior probability threshold.

Finally, we observed an average increase of 2.5% in valid prediction times when using InWiSe in comparison with a single OW in the size of N x $SubW_{size}$. This was mainly caused by the relaxed validation conditions in terms of missing values when splitting the windows, as well as by being able to use a single sub-window instead of a sub-window set at the beginning of a record when the available OWs are too short to be valid.

Cross-Datasets: The results of the cross-dataset experiments for the all-time prediction mode are shown in Table 1. First, one can observe the expected, but relatively small, drop in performance when training with one dataset and testing with another (0.1–0.5% in AUC). Nevertheless, we see that InWiSe outperforms other methods in terms of the AUC metric, even when applying the model to a new dataset. However, similarly to the in-dataset exclusive mode case, the other metrics domination in the cross-dataset validation (all-time mode) is threshold dependent, but this time with dependence on the source dataset. For example, the Soroka dataset sensitivity of the single OW method is higher than the sub-window set one (0.897 vs. 0.868, respectively) in the case where the model was trained on MIMIC II, while the opposite is true when it was trained on Hasassah (0.920 vs. 0.940). The reason for these results is probably the difference between the optimal threshold values in the source and the target datasets.

Feature Importance: The goal of splitting OWs was to let the classifier learn feature correlations with the target window from each sub-window separately, as well as their association and cross-window correlation. Table 3 presents the top

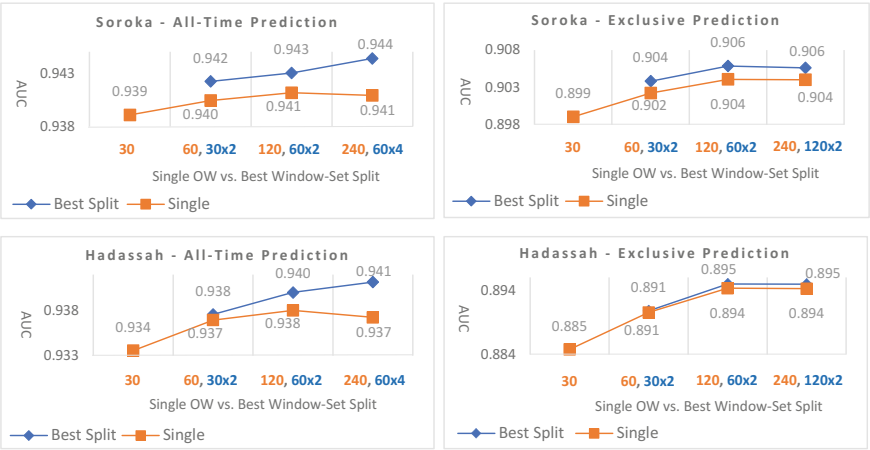

Fig. 3. In-dataset comparison between the single OW method and its InWiSe best split on the Hadassah and Soroka datasets (both prediction modes, XGBoost only).

Table 1. In-dataset and cross-dataset methods comparison (InWiSe best configuration vs. single OW best size vs. window-set matching single OW size) using all-time prediction. Metrics from top to bottom: AUC, accuracy, sensitivity, specificity and precision

Target		Source Dataset (Train)								
		MIMIC II			Soroka			Hadassah		
		4x60	120	240	4x60	120	240	4x60	120	240
MIMIC II	AUC	**0.937±0.003**	0.931±0.006	0.931±0.007	**0.933**	0.929	0.932	**0.937**	0.933	0.934
	Acc.	**0.878±0.007**	0.864±0.013	0.866±0.011	0.846	0.828	0.860	0.898	0.903	0.902
	Sens.	**0.849±0.011**	0.847±0.025	0.843±0.023	0.869	0.873	0.849	0.829	0.807	0.803
	Spec.	**0.879±0.008**	0.865±0.013	0.866±0.011	0.846	0.826	0.861	0.900	0.906	0.905
	Prec.	**0.168±0.007**	0.150±0.011	0.151±0.016	0.136	0.124	0.146	0.187	0.194	0.192
Soroka	AUC	**0.939**	0.935	0.935	**0.944±0.003**	0.941±0.003	0.941±0.003	**0.942**	0.938	0.939
	Acc.	0.867	0.822	0.858	**0.870±0.007**	0.861±0.005	0.856±0.004	0.761	0.800	0.793
	Sens.	0.868	0.897	0.865	**0.875±0.014**	0.875±0.011	0.875±0.014	0.940	0.915	0.920
	Spec.	0.867	0.820	0.857	**0.869±0.008**	0.860±0.006	0.856±0.007	0.754	0.796	0.789
	Prec.	0.187	0.150	0.178	**0.191±0.015**	0.182±0.016	0.177±0.017	0.119	0.137	0.134
Hadassah	AUC	**0.936**	0.933	0.932	**0.938**	0.935	0.935	**0.941±0.002**	0.938±0.002	0.937±0.002
	Acc.	0.842	0.816	0.815	0.838	0.814	0.834	**0.870±0.006**	0.867±0.007	0.861±0.006
	Sens.	0.889	0.899	0.896	0.894	0.904	0.888	**0.871±0.006**	0.866±0.005	0.866±0.006
	Spec.	0.839	0.812	0.811	0.835	0.809	0.831	**0.870±0.007**	0.867±0.007	0.860±0.007
	Prec.	0.207	0.184	0.182	0.204	0.183	0.198	**0.241±0.011**	0.235±0.011	0.226±0.010

Table 2. In-dataset methods comparison (InWiSe best configuration vs. single OW best size) in the exclusive prediction mode

	MIMIC II		Soroka		Hadassah	
	2×60	120	2×60	120	2×60	120
AUC	$\mathbf{0.884 \pm 0.011}$	0.884 ± 0.006	$\mathbf{0.906 \pm 0.002}$	0.904 ± 0.002	$\mathbf{0.895 \pm 0.003}$	0.894 ± 0.003
Accuracy	0.790 ± 0.020	$\mathbf{0.816 \pm 0.013}$	0.787 ± 0.011	$\mathbf{0.809 \pm 0.010}$	0.783 ± 0.009	$\mathbf{0.807 \pm 0.007}$
Sensitivity	$\mathbf{0.826 \pm 0.021}$	0.793 ± 0.023	$\mathbf{0.875 \pm 0.009}$	0.851 ± 0.012	$\mathbf{0.855 \pm 0.009}$	0.829 ± 0.007
Specificity	0.789 ± 0.021	$\mathbf{0.817 \pm 0.014}$	0.785 ± 0.010	$\mathbf{0.808 \pm 0.010}$	0.782 ± 0.009	$\mathbf{0.807 \pm 0.007}$
Precision	0.054 ± 0.006	$\mathbf{0.060 \pm 0.004}$	0.069 ± 0.004	$\mathbf{0.075 \pm 0.004}$	0.082 ± 0.003	$\mathbf{0.089 \pm 0.003}$

Table 3. Top 10 features rank in terms of frequency over trees in the XGBoost ensemble

	All-Time Prediction						Exclusive Prediction			
	60-min	240-min	4x60-min				60-min	120-min	2x60-min	
			$SubW_1$	$SubW_2$	$SubW_3$	$SubW_4$			$SubW_1$	$SubW_2$
SBP Slope	10	10	–	–	–	9	–	10	–	9
SBP Skewness	–	–	–	–	–	–	9	–	–	–
DBP Slope	–	–	–	–	–	–	–	–	–	10
MAP Mean	1	1	5	–	6	1	1	1	4	1
MAP Std	6	8	–	–	–	–	8	9	–	–
MAP Median	2	3	7	–	–	2	3	3	7	5
MAP Iqr	9	–	–	–	–	–	10	–	–	–
MAP Skewness	8	7	–	–	–	–	6	6	–	8
MAP Slope	4	2	–	–	–	3	5	4	–	3
HR Slope	5	6	–	–	–	8	4	5	–	6
SBP Cross Correlation w/ DBP	7	5	–	–	–	10	7	7	–	–
PP Cross Correlation w/ RCO	3	4	–	–	–	4	2	2	–	2
MAP Wavelet Detail Level-3	–	–	–	–	–	–	–	8	–	–
HR Wavelet Approximation	–	9	–	–	–	–	–	–	–	–

ten important features of XGBoost (most frequent over the ensemble trees) for the best InWiSe configuration compared with its sub-window sized OW as well as the corresponding single OW (in two prediction modes). The sub-window set columns are divided into their sub-windows, where $SubW_N$ is the sub-window ending at the prediction time and $SubW_1$ is the earliest in the set (Fig. 1b).

We first see that the Mean Arterial blood Pressure (MAP) mean is clearly dominant in all cases, which makes sense since MAP values are the ones that define an HE. Next, we observe that features from all three types (statistical, cross-correlation and wavelets) are top-ten-ranked, with the statistical ones (especially of MAP) used more frequently. Moreover, the two derived parameters, Pulse Pressure (PP) and Relative Cardiac Output (RCO), are proved to contribute particularly in their cross correlation with the target. Turning to sub-window sets, while $SubW_N$ has obviously more weight, the algorithm repeatedly chooses to combine the MAP mean and median from early sub-windows as well, with a surprisingly high rank. The early sub-window features are in favor of other higher ranked features in the single OWs (i.e., HR Slope and SBP cross-correlation with DBP). These findings support and explain our hypothesis that the model may be improved by using local sub-window features instead of extracting features from a single long OW.

5 Conclusions and Future Work

The current study presented and explored InWiSe, an enhanced feature extraction algorithm for clinical episode prediction, where physiological features are extracted from a set of observation sub-windows instead of a single OW. Our evaluation experiments have shown that the prediction performance may be improved by combining local sub-window features instead of extracting the same

features from a single OW (of any size), observing an increased improvement when splitting longer OWs than used in existing methods (i.e., 240-min OW). The importance of sub-window features is confirmed by a feature importance analysis. Moreover, in the all-time prediction mode, used by the recent works, we show an improvement in comparison with the single OW method over all three experimental datasets w.r.t. all evaluated metrics[2] (up to 1% in accuracy and specificity and up to 10% in precision, while maintaining the sensitivity equal or better). We particularly focus on the AUC metric that was improved by up to 0.6%, with a statistically significant improvement in AUC performance in the case of the Hadassah dataset.

In addition to the above, we successfully evaluated the methods in a cross-dataset fashion, showing that the induced models are capable of predicting episode on a new dataset, with just a little degradation in the performance. Moreover, the AUC metric repeatedly favors the InWiSe method, even when testing the model on a new dataset. Furthermore, we explored a new prediction mode (exclusive) which may better reflect ICU needs.

With regard to InWiSe future improvements, better accuracy results may be achieved in the case of an invalid sub-window set, especially in the exclusive prediction mode. One may also evaluate alternative approaches to multiple OW utilization such as the weighted ensemble method of [15]. Another possible approach to episode prediction may be built on predicting the Mean Arterial blood Pressure (MAP) values in the target window with multivariate time series forecasting models.

From the dataset perspective, any future analysis should use the recently published MIMIC III dataset mentioned in Sect. 2.5. Applying an existing model on a new dataset should be further investigated in terms of determining a dataset-specific optimal classification threshold. Finally, the proposed methodology can be extended to other episode prediction domains.

References

1. Ghosh, S., et al.: Hypotension risk prediction via sequential contrast patterns of ICU blood pressure. IEEE J. Biomed. Health Inform. **20**(5), 1416–1426 (2016)
2. Sebat, F., et al.: Effect of a rapid response system for patients in shock on time to treatment and mortality during 5 years. Crit. Care Med. **35**(11), 2568–2575 (2007)
3. Cao, H., et al.: Predicting ICU hemodynamic instability using continuous multi-parameter trends. In: Engineering in Medicine and Biology Society (EMBS), pp. 3803–3806. IEEE (2008)
4. Moody, G.B., Lehman, L.W.H.: Predicting acute hypotensive episodes: the 10th annual physionet/computers in cardiology challenge. In: Computers in Cardiology, pp. 541–544. IEEE (2009)
5. Forkan, A.R.M., et al.: ViSiBiD: a learning model for early discovery and real-time prediction of severe clinical events using vital signs as big data. Comput. Netw. **113**, 244–257 (2017)

[2] All improvement percentages are in terms of a ratio between the two measures

6. Kamio, T., et al.: Use of machine-learning approaches to predict clinical deterioration in critically Ill patients: a systematic review. Int. J. Med. Res. Health Sci. **6**(6), 1–7 (2017)
7. Eshelman, L.J., et al.: Development and evaluation of predictive alerts for hemodynamic instability in ICU patients. In: 2008 AMIA Annual Symposium Proceedings, p. 379. American Medical Informatics Association (2008)
8. Lee, J., Mark, R.G.: An investigation of patterns in hemodynamic data indicative of impending hypotension in intensive care. Biomed. Eng. Online **9**(1), 62 (2010)
9. Saeed, M., et al.: Multiparameter intelligent monitoring in intensive care II (MIMIC-II): a public-access ICU database. Crit. Car. Med. **39**(5), 952 (2011)
10. Chen, X., et al.: Forecasting acute hypotensive episodes in intensive care patients based on a peripheral arterial blood pressure waveform. In: 2009 Computers in Cardiology, pp. 545–548. IEEE (2009)
11. Saeed, M.: Temporal pattern recognition in multiparameter ICU data, Doctoral dissertation, Massachusetts Institute of Technology (2007)
12. Saeed, M., Mark, R.: A novel method for the efficient retrieval of similar multiparameter physiologic time series using wavelet-based symbolic representations. In: AMIA Annual Symposium Proceedings, p. 679. American Medical Information Association (2006)
13. Rocha, T., et al.: Wavelet based time series forecast with application to acute hypotensive episodes prediction. In: Engineering in medicine and biology society (EMBC), pp. 2403–2406. IEEE (2010)
14. Ghosh, S., et al.: Septic shock prediction for ICU patients via coupled HMM walking on sequential contrast patterns. J. Biomed. Info. **66**, 19–31 (2017)
15. Lee, J., Mark, R.G.: A hypotensive episode predictor for intensive care based on heart rate and blood pressure time series. In: Computing in Cardiology, pp. 81–84. IEEE (2010)
16. Rocha, T., et al.: Prediction of acute hypotensive episodes by means of neural network multi-models. Comp. Biol. Med. **41**(10), 881–890 (2011)
17. Goldberger, A.L., et al.: PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation **101**(23), e215–e220 (2000)
18. The MIMIC II Waveform Database Matched Subset (Physionet Database). https://physionet.org/physiobank/database/mimic2wdb/matched/
19. Lee, Y.-L., Juan, D.-C., Tseng, X.-A., Chen, Y.-T., Chang, S.-C.: DC-Prophet: predicting catastrophic machine failures in DataCentre. In: Altun, Y., et al. (eds.) Machine Learning and Knowledge Discovery in Databases. LNCS, vol. 10536, pp. 64–76. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-71273-4_6
20. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: 22nd ACM SIGKDD International Conference, pp. 785–794. ACM (2016)