Published in final edited form as:

Inform Med Unlocked. 2019 ; 16: . doi:10.1016/j.imu.2019.100222.

Signal quality measure for pulsatile physiological signals using morphological features: Applications in reliability measure for pulse oximetry

Elyas Sabeti[f,b,*], **Narathip Reamaroon**[a], **Michael Mathis**[e,b], **Jonathan Gryak**[a], **Michael Sjoding**[c,b], **Kayvan Najarian**[a,d,b]

[a]Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, 48109, USA

[b]Michigan Center for Integrative Research in Critical Care (MCIRCC), University of Michigan, Ann Arbor, MI, 48109, USA

[c]Department of Internal Medicine, University of Michigan, Ann Arbor, MI, 48109, USA

[d]Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, 48109, USA

[e]Department of Anesthesiology, University of Michigan, Ann Arbor, MI, 48109, USA

[f]Michigan Institute for Data Science, University of Michigan, Ann Arbor, MI, 48109, USA

Abstract

Pulse oximetry is a noninvasive and low-cost physiological monitor that measures blood oxygen levels. While the noninvasive nature of pulse oximetry is advantageous, the estimates of oxygen saturation generated by these devices are prone to motion artifacts and ambient noise, reducing the reliability of such estimations. Clinicians combat this by assessing the quality of oxygen saturation estimation by visual inspection of the photoplethysmograph (PPG), which represents changes in pulsatile blood volume and is also generated by the pulse oximeter. In this paper, we propose six morphological features that can be used to determine the quality of the PPG signal and generate a signal quality index. Unlike many similar studies, this approach uses machine learning and does not require a separate signal, such as ECG, for reference. Multiple algorithms were tested against 46 30-min PPG segments of patients with cardiovascular and respiratory conditions, including atrial fibrillation, hypoxia, acute heart failure, pneumonia, ARDS, and pulmonary embolism. These signals were independently annotated for signal quality by two clinicians, with the union of their annotations used as the ground-truth. Similar to any physiological signal recorded in a clinical setting, the utilized dataset is also unbalanced in favor of good quality segments. The experiments showed that a cost-sensitive Support Vector Machine (SVM) outperformed other tested methods and was robust to the unbalanced nature of the data. Though the proposed

[*]Corresponding author. Michigan Institute for Data Science (MIDAS), University of Michigan, Ann Arbor, MI, 48109, USA. sabeti@umich.edu (E. Sabeti).

Appendix A. Supplementary data
Supplementary data to this article can be found online at https://doi.org/10.1016/j.imu.2019.100222.

algorithm was tested on PPG signals, the methodology remains agnostic to the dataset used, and may be applied to any type of pulsatile physiological signal.

Keywords

Pulsatile physiological signal; Signal quality index; Pulse oximetry; PPG; Cost-sensitive SVM; Decision tree ensemble; ARDS

1. Introduction

Pulsatile physiological signals are often noninvasive recordings of blood-related physiological measurements used in health monitoring. The quality of these recordings is a major concern in healthcare [1,2], as many vital physiological measurements (e.g., respiratory rate, heart rate, and oxygen saturation) are extracted from these signals. The pulsatile nature and similarity of patterns across these signals [3] makes it possible to develop a general algorithm for quality assessment. Additionally, due to the optical sensors used for noninvasive recording of pulsatile signals, the prominent noise sources contaminating these signals are also the same, i.e., motion artifacts and ambient light [3]. Thus, in this paper six morphological features and a machine learning framework are introduced in order to measure the quality of any pulsatile physiological signal and detect segments of poor quality. As an initial application, the algorithm was tested on PPG signals generated from pulse oximeters.

1.1. Pulse oximetry

Pulse oximetry is a low-cost noninvasive tool that has been widely employed in healthcare to extract vital signs such as oxygen saturation and heart rate from the recorded pulsatile signal. Arterial oxygen saturation (SaO_2) is defined as the ratio of oxygenated hemoglobin to the combined amount of oxygenated and deoxygenated hemoglobin present in arterial blood, and it is indicative of cardiovascular and respiratory status. A photoplethysmograph (PPG), which represents changes in pulsatile blood volume signal, is recorded by the pulse oximeter and is used in estimating the oxygen saturation non-invasively. The estimated value of SaO_2 using pulse oximetry is called peripheral oxygen saturation (SpO_2). Currently pulse oximeters use a weighted average method to compute values of SpO_2, however this estimate is prone to many artifacts including ambient light, motion, and those due to low perfusion [3]. Thus, the reliability of SpO_2 is highly dependent upon the quality of the PPG signal. Signal quality is becoming more relevant due to the increasing use of telemedicine, as well as the need to reduce false alarms in intensive care units [1,2]. Additionally, studies have shown that medical data quality affects the performance of clinical decision support systems significantly [4–7].

In the literature there are several studies of pulsatile signal quality (especially PPG signal), but many of these works use either an incomplete dataset, other simultaneous signals, or did not take a machine learning approach. For instance, in Ref. [8] a novel online algorithm based on gradient ascent was proposed to estimate the quality of pulsatile signals. However, the ECG signal was used as an input (and reference) of their algorithm.

In Ref. [9] four morphological and temporal features are extracted and used as states in a Kalman filter to adaptively accept or reject signals based on their quality. In Ref. [10], a novel method using repeated Gaussian filters for localizing segments of pulses was developed, with cross-correlation of consecutive pulse segments used to calculate a signal quality index. However, their proposed algorithm has no learning process and the threshold on SQI is calculated experimentally. In Ref. [11] three SQI algorithms are proposed to analyze the effect of motion artifact on PPG signal, with only one of them solely relying on PPG signal while the other two rely on red and infrared signals. In Ref. [12], authors defined an SQI and focused on reliability of heart rates obtained from ECG and PPG collected using wearable sensors. Their proposed SQI algorithm is essentially a cascade of decision rules on RR intervals and heart rate combined with adaptive template matching. In Ref. [13] kurtosis and Shannon entropy were used in a statistical framework for detecting motion and noise artifact with multi-site (i.e., ear, finger, and forehead) PPG signals. In each segment of PPG, the kurtosis and entropy are compared with thresholds and their fusion is used as a metric for artifact detection. It was concluded that forehead and finger sensors have the highest and lowest contamination respectively. In Ref. [14], a framework was proposed in which PPG beats are detected and their quality estimated by comparing the beats with a template. For each beat a quality index is calculated using the normalized root mean squared error of each beat with respect to the template derived from the surrounding pulses using dynamic time warping barycenter averaging. The principle drawback of the proposed algorithm is its limitation to offline settings. In Ref. [15], an algorithm was presented in which beats are first localized, after which they are resampled to enforce equal beat duration, with beat quality estimated by calculating the similarity between consecutive beats. In this framework, spline interpolation is used as the resampling method to ensure the equality of beat duration, and the Pearson correlation coefficient is used to measure the similarity between the two consecutive resampled PPG beats. The classification in this method relies on a threshold that is determined by enforcing the quality of sensitivity and specificity of training data. In Ref. [16], a real-time PPG quality assessment is presented with focus on reduction of the energy consumption and false alarms. The proposed method determines the quality of a PPG segment in a four-step hierarchical decision-making process, with each step comparing a feature against a threshold. The features used are maximum absolute amplitude, local amplitude maxima, the zero-crossing rate, and autocorrelation. The potential limitations of the aforementioned approaches include not employing any machine learning method and reliance upon empirical thresholds that determine the quality of PPG segments.

In many pulsatile signal quality assessment methodologies, morphological or Signal Quality Index (SQI)-based features are extracted and utilized by machine learning algorithms. In Ref. [17] dynamic time-warping (DTW) is first used to align each beat to match a running template, after which four signal quality metrics are extracted. In their best performing method, the four signal quality metrics, a fusion SQI, and the number of beats are presented to a multilayer perceptron (MLP) neural network. The algorithm’s performance was tested on an expert-labeled database of 1055 six-seconds segments of PPG. The weights of the trained MLP are specific to the type of data on which it is trained, requiring re-training in order to assess the quality of each type of pulsatile signal. In Ref. [18] the author developed a signal quality algorithm for PPG using eight SQIs, including perfusion, kurtosis, and skewness. Four classifiers were tested to classify the 106 1-min recordings into excellent, acceptable, or unfit, with skewness yielding the best performance. In Ref. [19], PPG signals of patients with atrial fibrillation were divided into segments of 30 s, after which 42 temporal and spectral features (e.g., mean, median, standard deviation, Shannon entropy, median frequency, and spectral entropy) were extracted from each segment. Three machine learning methods (support vector machine, *k*-nearest neighbors, and decision tree) were then investigated, with support vector machine outperforming the other two methods.

In studies such as [1,2,20], multiple intensive care unit signals were considered to estimate the signal quality for false alarm reduction. More specifically, in Ref. [1] a relevance vector machine (RVM) was trained with 114 physiological and signal quality features extracted from the ECG, PPG, and arterial blood pressure waveforms to reduce false alarms in an intensive care unit (ICU). In Ref. [2] a temporal vector of samples from many waveforms including respiration waveform, PPG, and multiple ECG leads were used in an adaptive filtering and prediction algorithm called MCAF to generate point-by-point SQI. In Ref. [20] a supervised machine learning algorithm was used to classify alerts as real or artifacts in online noninvasive vital sign data streams (heart rate, respiratory rate, peripheral oximetry) to reduce alarm fatigue and missed true instability.

The proposed method in this paper is primarily related (in terms of feature space) to Ref. [21], in which morphological features such as pulse amplitude, trough depth difference between successive pulse troughs, and pulse width are used in conjunction with multiple heuristic thresholds to divide 104 1-min fingertip PPG signals into good and artifact signals. The result of their proposed algorithm was then compared with expert-generated labels. This approach is limited by its use of heuristics and its need for a simultaneous ECG signal for reference.

To overcome the limitations of prior works, in this paper an automatic machine learning framework is designed based on features extracted solely from the PPG signal to classify beats and intervals as good or poor quality. A cohort of patients with hypoxia, acute heart failure, pneumonia, acute respiratory distress syndrome (ARDS) and other respiratory conditions was created for this study. PPG signals from this cohort were manually annotated by two clinicians to produce a gold standard, to which the performance of the algorithm was compared. Additionally, the proposed algorithm is also tested against a public dataset.

2. Material

The primary dataset used in this study is part of an ongoing ARDS project, consisting of patients treated at Michigan Medicine. In an effort to compare the proposed algorithm to published methods, the publicly available dataset CapnoBase was used. However, the primary focus of this paper is on the ARDS database.

2.1. ARDS dataset

As a part of ongoing research at Michigan Medicine, a retrospective cohort of adult patients hospitalized between November 2016 and June 2017 with moderate hypoxia, acute heart failure, pneumonia, acute respiratory distress syndrome (ARDS) and other respiratory conditions was created. All these patients required mechanical ventilation during the first 7 days of their hospitalization. We acquired data from bedside telemetry monitors of all patients, which is currently stored in the Michigan Center for Integrative Research in Critical Care (MCIRCC) Databank. The PPG recording equipment used in this study is Masimo LNCS DCI adult reusable sensor with GE Medical PDM interface. The sampling frequency of the PPG signals in the dataset is 60 Hz.

For the current pulse oximetry quality study, 46 30-min segments of PPG signal from different patients with various cardiovascular and respiratory conditions including atrial fibrillation, hypoxia, acute heart failure, pneumonia, ARDS, and pulmonary embolism were extracted. 27 (out of 46) of these patients are male, the average age of the patients is 57 years old, and 37 (out of 46) are Caucasian. Among these 46 30-min segments, only 12 segments are almost entirely normal, 20 segments contain long episodes of atrial fibrillation and sinus tachycardia, and the rest contain sporadic short-term abnormalities (finger tapping, premature atrial contractions and etc). Two clinicians independently reviewed PPG signals for uniform, pulsatile changes in the waveform, based on their experience interpreting such waveforms in clinical settings. Waveforms without a clear pulsatile signal (regardless of arrhythmic episodes, only based on morphology) that a clinician would not have trusted as accurate in a clinical setting were annotated as poor quality segments. Also, certain pulsatile waveforms suspicious for artifact, e.g., finger tapping, were also annotated as poor quality. Fig. 1 illustrates a 24-second segment of PPG signal annotated for signal quality by both clinicians. The union of their labels is used as ground-truth for the algorithm. This cohort is primarily used in order to develop and validate the proposed algorithms.

2.2. CapnoBase dataset

The CapnoBase (CB) dataset [10] consists of 42 8-min finger transmissive PPG recordings (29 pediatric, 13 adult) collected during elective surgery and routine anesthesia with a sampling frequency of 300 Hz. This dataset also includes signal annotations adjudicated by a research assistant.

3. Methodology

In this section, an overview of signal annotation, the proposed algorithm, and validation framework is provided. The advantages of the proposed methodology are three-fold: 1) applicability to any source of pulsatile physiological signals; 2) independence from any synchronized reference signal; 3) adaptivity to any dataset. As discussed below, the adaptivity of the proposed framework to any pulsatile physiological signal or any dataset is due to the normalization within the defined morphological features and the auto-calibration of the algorithm to the non-artifact changes of the signal. This approach can be applied to any source of pulsatile waveform; thus, in this section it is assumed that the signal under assessment is PPG.

3.1. Algorithm development and features

One of the advantages of the algorithm is that it only uses PPG, and no other synchronized signal such as ECG is needed. Additionally, it is fully automated. The input of the algorithm is a PPG signal and its sampling frequency, and the output is a classification of segments into “good” or “poor” quality. For any interval, the output can be a number between 0 and 1 that can be construed as a signal quality index (SQI).

3.1.1. Preprocessing and calibration—

In the preprocessing phase, a raw PPG signal is first filtered using a band-pass Butterworth filter with a 0.5–5 Hz pass band as suggested in Ref. [21]. The next step is peak detection, wherein potential peaks are only considered if the minimum temporal distance between two consecutive beats is 70% of the mean PPG beat period. Heart rate is adaptively extracted from the power spectrum of the most recent 20 s of PPG signal, as the frequency between 1 Hz and 3 Hz having maximum power spectrum determines the heart rate.

Unlike many algorithms in the literature that use ECG a reference for beat detection (e.g., Ref. [21]),the proposed algorithm is independent of any other signal. Moreover, positive and negative peaks are detected independently, resulting in two heart rate signals that should be approximately the same. As described later, the difference between these two heart rate signals is used as a feature of the algorithm, for any significant dissimilarity is due to abnormality in beat morphology. Fig. 2 depicts examples of raw and filter PPGs with detected positive and negative peaks.

3.1.2. Morphological features—

First, six morphological signals/measurements are extracted that are used later to extract features (Fig. 2): (1) beat waveform with positive peak (the interval between two negative peaks), (2) beat waveform with negative peak (the interval between two positive peaks), (3) change in absolute amplitude between two consecutive negative peaks, (4) change in absolute amplitude between two consecutive positive peaks, (5) heart rates extracted from positive peaks and negative peaks (or pulse width) and (6) absolute positive to negative peak amplitude, i.e., the AC component. The next step is to use the extracted signals/measurements to calculate morphological features. All of the proposed features are based on some distance or dissimilarity from baseline values or templates. One can think of these templates and baseline values as adaptive averages extracted from normal beats/signals that have already been seen. For now, assume the algorithm is provided with these adaptive averages and focus on the features; later it is described how these averages can be calculated.

Let f_s be the sampling frequency and suppose $\mathscr{T} = \left\{ t_k \middle| k \in \mathbb{N},\ t_k = k\frac{1}{f_s} \right\}$ is the set of time samples in the PPG signal. Assume $f_{\mathrm{PPG}}: \mathscr{T} \to \mathscr{V}$ is the PPG signal amplitude function and $\mathscr{V}$ is the bounded set of these amplitude values, i.e. $\mathscr{V} = \{ v_k | k \in \mathbb{N},\ t_k \in \mathscr{T},\ v_k = f_{\mathrm{PPG}}(t_k) \in \mathbb{R} \}$. The features are then extracted as follows:

3.1.2.1. Normalized pulse duration.:

Suppose $\mathscr{P}^+$ and $\mathscr{P}^-$ are respectively the set of positive and negative peak locations defined as

$$\mathscr{P}^+ = \left\{ p_i^+ \middle| i \in \mathbb{N},\ p_i^+ \in \mathscr{T} : \forall t \in [p_{i-1}^-,\ p_i^-] \subseteq \mathscr{T},\ f_{\mathrm{PPG}}\left(p_i^+\right) \geq f_{\mathrm{PPG}}(t) \right\}$$

$$\mathscr{P}^- = \left\{p_i^- \,|\, i \in \mathbb{N},\ p_i^- \in \mathscr{T} : \forall t \in \left[p_{i-1}^+,\ p_i^+\right] \subseteq \mathscr{T},\ f_{\text{PPG}}(p_i^-) \le f_{\text{PPG}}(t) \,.\right\}$$

Then for each consecutive pair of positive peaks $\left(p_{i-1}^+,\ p_i^+\right) \in \left(\mathscr{P}^+\right)^2$ or negative peaks $(p_{i-1}^-,\ p_i^-) \in (\mathscr{P}^-)^2$ define the **normalized pulse duration,** $\overline{\nabla p_i}$, as

$$\overline{\nabla p_i} = \frac{\nabla p_i - \nabla p}{\nabla p},$$

where

$$\nabla p_i = p_i - p_{i-1} = \begin{cases} p_i^+ - p_{i-1}^+ & (p_{i-1},\ p_i) \in \left(\mathscr{P}^+\right)^2 \\ p_i^- - p_{i-1}^- & (p_{i-1},\ p_i) \in (\mathscr{P}^-)^2 \end{cases}$$

and ∇p is the baseline value (as defined in section 3.1.3) of pulse duration. An example of ∇p_i can be seen in Fig. 2. Given that for every interval between two consecutive positive (negative) peaks there is a negative (positive) peak, each value of $\overline{\nabla p_i}$ is only associated with the interval between the first positive (negative) peak to the next negative (positive) peak.

3.1.2.2. Normalized negative-to-negative peak jump.: Define the set $\mathscr{A}^- = f_{\text{PPG}}(\mathscr{P}^-) = \{P_i^- \,|\, i \in \mathbb{N}, \forall p_i^- \in \mathscr{P}^-,\ P_i^- = f_{\text{PPG}}(p_i^-)\}$ as the set of negative peak amplitudes, and let ∇P^- be the baseline value for negative-to-negative peak jump and ∇P be the baseline value for amplitude change from negative to positive (or positive to negative) peaks, i.e., the baseline value for the AC component. For each pair of consecutive negative peaks $(p_{i-1}^-, p_i^-) \in (\mathscr{P}^-)^2$, the **normalized negative-to-negative peak jump,** $\overline{\nabla P_i^-}$, is defined as

$$\overline{\nabla P_i^-} = \frac{\nabla P_i^- - \nabla P^-}{\nabla P},$$

where $\nabla P_i^- = |P_i^- - P_{i-1}^-|$.

3.1.2.3. Normalized positive-to-positive peak jump.: Similar to previous section, suppose $\mathscr{A}^+ = f_{\text{PPG}}\left(\mathscr{P}^+\right) = \left\{P_i^+ \middle| i \in \mathbb{N}, \forall p_i^+ \in \mathscr{P}^+, P_i^+ = f_{\text{PPG}}\left(p_i^+\right)\right\}$ is the set of positive peak amplitudes and ∇P^+ is the baseline value for positive-to-positive peak jump. For each pair of consecutive positive peaks $\left(p_{i-1}^+, p_i^+\right) \in \left(\mathscr{P}^+\right)^2$, the **normalized positive-to-positive peak jump,** $\overline{\nabla P_i^+}$, is defined as

$$\overline{\nabla P_i^+} = \frac{\nabla P_i^+ - \nabla P^+}{\nabla P},$$

where $\nabla P_i^+ = \left| P_i^+ - P_{i-1}^+ \right|$.

3.1.2.4. Normalized beat amplitude jump.: Suppose $\mathscr{P} = \mathscr{P}^- \cup \mathscr{P}^+$ is the set of positive and negative peak locations and $\mathscr{A} = \mathscr{A}^- \cup \mathscr{A}^+$ is the set of peak amplitudes. Then for any consecutive positive and negative peak $\left(p_{i-1},\ p_i\right) \in \mathscr{P}^2$, the **normalized beat amplitude jump,** $\overline{\nabla P_i}$, is defined as

$$\overline{\nabla P_i} = \frac{\nabla P_i - \nabla P}{\nabla P},$$

where $\nabla P_i = |P_i - P_{i-1}|$.

3.1.2.5. Dissimilarity measure of positive-peaked beats.: As described in Ref. [17], due to nonlinear and non-stationary changes in beat morphology, a nonlinear time-based stretching or compression of beats is necessary to perform effective template matching. As mentioned earlier in this section, beat waveforms with positive peak (interval between two negative peaks, see Fig. 2) are extracted and normalized into the range [0,1]. Then, dynamic time warping (DTW) is used to align the PPG with a template as constructed in section. 3.1.3. A brief description of DTW algorithm for PPG is provided in Ref. [17]. Finally, KL divergence [22] is used to measure the difference between the aligned PPG beat and the template, which is formulated as

$$D\left(T^+ \parallel B^+\right) = \sum_{i=1}^{m} t_i^+ \log \frac{t_i^+}{b_i^+},$$

where $B^+ = \left\{ b_k^+ \middle| 1 \le k \le m \right\}$ and $T^+ = \left\{ t_k^+ \middle| 1 \le k \le m \right\}$ are two aligned time series of beats and template with positive peak, both of which are of length m and normalized such that $\sum_{i=1}^{m} b_i^+ = \sum_{i=1}^{m} t_i^+ = 1$. In the proposed algorithm, $D(T^+ \parallel B^+)$ is used as the **dissimilarity measure of positive-peaked beats** feature.

3.1.2.6. Dissimilarity measure of negative-peaked beats.: Applying the same procedure as described above, a **dissimilarity measure of negative-peaked beats,** i.e. $D(T^- \parallel B^-)$, is calculated in which $B^- = \{b_k^- | 1 \le k \le m\}$ and $T^- = \{t_k^- | 1 \le k \le m\}$ are two time series of beat and template with negative peak, both of which are of length k and normalized such that $\sum_{i=1}^{k} b_i^- = \sum_{i=1}^{k} t_i^- = 1$.

3.1.3. Templates and baseline values—

As described in section 3.1.2, the proposed features $D(T^- \parallel B^-)$ and $D(T^- \parallel B^-)$ require templates, while the features $\overline{\nabla p_i}$, $\overline{\nabla P_i^-}$, $\overline{\nabla P_i^+}$, and $\overline{\nabla P_i}$ need baseline values. One of the distinct components of the proposed algorithm is that these templates and values are generated individually for each waveform. In this section it is described how to generate these templates.

3.1.3.1. Initial template and baseline value generation.: Our proposed algorithm uses the first T seconds of each waveform in the calibration phase, during which preprocessing and then peak detection is performed on the segment. Based on this segment and the peak locations, the baseline value ∇p is the averaged pulse duration, ∇P^- the average negative-to-negative peak jumps, ∇P^+ the average positive-to-positive peak jumps, and ∇P the average amplitude change from negative to positive peaks. In the results presented in this paper, $T =$ 20 seconds is chosen.

Formally, suppose $\mathscr{P}^{+}_{0+20} = \left\{p_i^+ \middle| i \in \mathbb{N}, p_i^+ \in \mathscr{T}, 0 < p_i^+ < 20\right\}$ and $\mathscr{P}^{-}_{0-20} = \{p_i^- | i \in \mathbb{N}, p_i^- \in \mathscr{T}, 0 < p_i^- < 20\}$ are the sets of positive and negative peak locations in the [0-20] time inteival, and assume there are m positive and m negative peaks in the 20 s segment, i.e., $\left|\mathscr{P}^{+}_{0-20}\right| = |\mathscr{P}^{-}_{0-20}| = m$ (the procedure is the same if the number of positive and negative peaks are not equal). Similarly, $\mathscr{A}^{+}_{0-20} = f_{\mathrm{PPG}}\left(\mathscr{P}^{+}_{0-20}\right)$ and $\mathscr{A}^{+}_{0-20} = f_{\mathrm{PPG}}(\mathscr{P}^{-}_{0-20})$. Also $\mathscr{P}_{0-20} = \mathscr{P}^{-}_{0-20} \cup \mathscr{P}^{+}_{0-20}$ and $\mathscr{A}_{0-20} = \mathscr{A}^{-}_{0-20} \cup \mathscr{A}^{+}_{0-20}$ are the sets of all (positive and negative) peaks and their amplitudes in the 20-second segment. Then

$$\nabla P^{+} = \frac{1}{m-1}\sum_{i=2}^{m}\left|f_{\mathrm{PPG}}\left(p_i^{+}\right) - f_{\mathrm{PPG}}\left(p_{i-1}^{+}\right)\right|$$

$$\nabla P^{-} = \frac{1}{m-1}\sum_{i=2}^{m}\left|f_{\mathrm{PPG}}(p_i^{-}) - f_{\mathrm{PPG}}(p_{i-1}^{-})\right|$$

$$\nabla P = \frac{1}{2m-1}\sum_{i=2}^{2m}\left|f_{\mathrm{PPG}}(p_i) - f_{\mathrm{PPG}}(p_{i-1})\right|$$

are the initial baseline values that will be used in the proposed algorithm. The value of ∇p is calculated based on the power spectrum of the 20 s segment, as the frequency between 1 and 3 Hz that has the highest power is the inverse of the heart rate frequency [3], i.e., $\frac{1}{\nabla p}$.

In order to extract an initial template with positive peak T^+, first the $m-1$ positive-peaked pulses are sorted with respect to their pulse width. If the template pulse duration is chosen to be the *mode* of pulse duration (the most frequent pulse duration) in the 20-second segment, then the template T^+ can be calculated as the average of beats that have the same temporal duration as the mode of pulse duration. If the mode of pulse duration is not unique, the *median* of pulse duration (the middle value for pulse duration) in the 20-second segment is chosen, and then the beats that have the width closest to the median of pulse duration will be aligned (e.g., by using DTW) or interpolated and then averaged to achieve the template T^+. The same procedure is applied to negative-peaked beats in order to extract the template with negative peak T^-.

3.1.3.2. Updating template and baseline values.: As mentioned above, the first T seconds of each waveform is used as the calibration phase to extract initial individual-specific templates and baseline values, with $T = 20$ seconds chosen for this paper. Since it's possible that the first segment is noisy, the initial baseline values and templates may be invalid. Thus, two criteria for accepting a segment as valid are imposed:

1. The number of positive or negative peaks should be more than $0.95 \times T$; i.e., on average a heartbeat should occur at least every 0.95 s.
2. At least one third of pulse widths (pulse durations) are within 5% of pulse duration mode/median (as mentioned in the previous section, if the mode of pulse duration is not unique, the median of pulse duration is chosen for template width).

If both of these conditions are satisfied, the first T seconds are accepted for initial baseline values extraction, otherwise the T-second window is iteratively slid for 1 s until both conditions are satisfied (e.g., intervals of [0,20] [1,21], etc.).

Due to the non-stationary nature of the source, after the initial calculation of the baseline values and templates, an adaptive algorithm for updating these baseline values and templates is necessary, particularly if the PPG signal has long duration. As such, after calculating the features of each segment using the baseline values and templates of the previous segment, these baseline values and templates are then updated to be used in the subsequent segment. Similarly, an interval is accepted for updating the baseline values and templates if it also satisfies the two aforementioned criteria.

3.1.4. A simple algorithm—Through feature extraction, each sample is represented as $\mathbf{x} \in \mathbb{R}^6$

$$\mathbf{x} = \begin{bmatrix} \overline{\nabla p} \\ \overline{\nabla P^{-}} \\ \overline{\nabla P^{+}} \\ \overline{\nabla P} \\ D\left(T^{+} \| B^{+}\right) \\ D\left(T^{-} \| B^{-}\right) \end{bmatrix}$$

where $\overline{\nabla p}$, $\overline{\nabla P^{-}}$, $\overline{\nabla P^{+}}$, $\overline{\nabla P}$, $D(T^{+} \| B^{+})$ and $D(T^{-} \| B^{-})$ are the features described in section 3.1. Using a simple algorithm based on decision rules, these values can be compared with thresholds for classification purposes. The hypothetical thresholds for a simplistic algorithm can be achieved experimentally using training data. After choosing the thresholds, the six features can be used to classify each beat, more specifically each interval between any two peaks (positive to negative peaks or negative to positive peaks), into a good or poor quality interval. In this case, a simple algorithm assigns the "poor quality" label to each interval between two consecutive peaks if any of the six features are greater than the threshold. Formally speaking, for an interval set $\mathcal{I}_{p_{i-1}}^{p_i} = [p_{i-1}, p_{\mathrm{i}}]$ between any two consecutive peaks (note that $\mathcal{I}_{p_i-1}^{p_i} \subseteq \mathcal{T}$), define the feature function $f_{\text{feature}} : \mathcal{T} \times \mathcal{V} \to \mathbb{R}^6$ as a function from the set of time samples and bounded set of PPG values to the feature space. The set of poor quality intervals is then

$$\mathcal{T}^{\text{Poor}} = \Big\{ t_j \,\Big|\, \exists \mathcal{I}_{p_i-1}^{p_i} \subseteq \mathcal{T} \text{ s.t. } t_j \in \mathcal{I}_{p_i-1}^{p_i}, \\ f_{\text{feature}}\left(\mathcal{I}_{p_i-1}^{p_i}, f_{\text{PPG}}\left(\mathcal{I}_{p_i-1}^{p_i}\right)\right) \nless \boldsymbol{\tau} \Big\},$$

where $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_6] \in \mathbb{R}^6$ is a vector of thresholds on the six features, and the inequality is performed component-wise. Obviously the set of good quality signal is the complement of $\mathcal{T}^{\text{Poor}}$, i.e. $\mathcal{T}^{\text{Good}} = \left(\mathcal{T}^{\text{Poor}}\right)^C = \mathcal{T} - \mathcal{T}^{\text{Poor}}$. This algorithm is used later for threshold optimization as described in section 4.1.4.

3.1.5. Interval classification and signal quality index—

One of the primary reasons for measuring PPG quality and reliability is that other important signals such as oxygen saturation utilize PPG in their formation. In general, oxygen saturation values are averaged over a moving window of PPG signal. In the ARDS dataset, the pulse oximetry hardware (PPG recording device) calculates every value of oxygen saturation based on the last 8 s of PPG signal. Consequently, having isolated poor quality beats/intervals is insufficient to label a PPG segment as “poor quality.” Thus, for an interval $\mathcal{T}_{t_k}$ of length 8 s such that $\mathcal{T}_{t_k} = \{t_i | t_i \in \mathcal{T}, t_i - 8 < t_i \le t_k\}$, the signal quality index (SQI) for that window is defined as

$$\text{SQI}\left(\mathcal{T}_{t_k}\right) = 1 - \frac{\left|\mathcal{T}_{t_k} \cap \mathcal{T}^{\text{Poor}}\right|}{\left|\mathcal{T}_{t_k}\right|} = \frac{\left|\mathcal{T}^{\text{Good}}\right|}{\left|\mathcal{T}_{t_k}\right|}, \qquad (1)$$

which is always a number between zero and one. As discussed in section 3.4, the SQI for any given interval will be compared with a pre-determined rate (threshold) for classification.

3.2. Learning models and decision rules

In this paper, two different training/testing frameworks are considered: (a) A standard learning method in which a single model is trained on 6-dimensional samples (Fig. 4a), and (b) six similar models that are trained on each sample feature separately, followed by a decision rule (Fig. 4b). The principle reason for considering the second model is the nature of the proposed normalized features, i.e., for a normal PPG beat *all* the features are expected to be close to zero; while for a poor quality interval, the absolute value of *at least* one of these features is expected to be greater than a threshold. To support this argument, Fig. 3 represents the cumulative distribution function (CDF) of the absolute value of the normalized negative-to-negative peak jump feature $\overline{\left|\nabla P_i^{-}\right|}$ for both classes. This figure shows that the larger the value of $\overline{\left|\nabla P_i^{-}\right|}$, the worse the quality, and this is valid for all the features. Thus, in the second framework, the decision rule is simply a logical “or” operation on the outcomes of each trained model on individual features.

3.3. Beat-scale analysis

One challenging aspect of the ARDS dataset is that the algorithms assign labels to each interval between consecutive peaks (positive peak to negative peak or negative peak to positive peak), while experts assign "poor quality" labels to any interval of any length – not necessarily to the beats. To perform beat-scale analysis, the signal annotations must be converted into beat-scale labels. If any subsequence of an interval between consecutive peaks is included in a segment annotated as poor quality segment, the label of that interval is "poor", otherwise it is labeled "good". Another challenge of the dataset – common to many medical datasets – is the unbalanced (also refer to as imbalanced [23–27]) proportion of class samples, i.e. there are far fewer poor quality features, compared to those of good quality. In fact, only about 5% of samples are of poor quality. Table 1 summarizes the percentage of poor quality samples in the dataset based on two experts annotations, their union and intersection. In this study, the union of labels is used as ground-truth.

3.4. Fixed interval-scale analysis

As mentioned in Section 3.1.5, the values of oxygen saturation in the dataset represent an average over fixed intervals of 8 s. In light of this, it's needed to determine what percentage of the 8 s interval must be considered poor quality before deeming the entire interval as poor. A rate parameter may be used in the analysis as a threshold for the SQI (equation (1) in each interval of 8 s. For any value of this parameter, an interval is of poor quality if its SQI is greater than the rate. Hence, by changing the rate from 0 to 1, sensitivity and specificity of any algorithm and the inter-rater reliability of expert annotations can be calculated. Fig. 5 illustrates the inter-rater reliability for annotations using Cohen's Kappa against the aforementioned rate. As can be seen, for a fixed interval of length 8 s, changing the rate does not have a significant impact on inter-rater reliability. Consequently, a rate of 0.5 is used on the union of annotation labels to determine the ground-truth label of these fixed intervals, since for this rate Cohen's Kappa is maximal.

4. Experiments

Using the ARDS dataset, 100 iterations of random subsampling is performed at the patient level. In each iteration the dataset is randomly divided into $\frac{2}{3}$ training data (31 30-min signals) and $\frac{1}{3}$ testing data (15 30-min signals). This results in a total of 234,739 samples at the beat-scale level. While the results of beat-scale analysis are also provided, we prioritize the performance of the proposed algorithm on intervals of fixed length (fixed interval-scale analysis). Though seemingly counter-intuitive, this approach can be considered sound, as physiological signals such as oxygen saturation are extracted based on the average value in fixed-length PPG segments. Hence, the reliability of these values depends on the quality of the fixed-length PPG segments. As mentioned earlier, another challenge of this dataset is the unbalanced nature of the data, hence learning methods such as a standard support vector machine (SVM) cannot be directly applied. As such, certain modifications are needed. In this study, for the first learning framework (Fig. 4a), SVM and an ensemble of trees are used; while for the second framework (Fig. 4b) a decision tree and a proposed learning method called *threshold optimization* are employed, which when combined with a non-uniform undersampling approach fits the feature space well. Fig. 6 represent the block diagram of the learning process.

4.1. Results

In this section, each learning method (decision tree, the ensemble of decision tree, support vector machine, and threshold optimization) is briefly described and their performance results with respect to both beat-scale and fixed interval-scale analyses are provided. Each result is the average of multiple simulations, each with different randomly generated training (31/46) and testing (15/46) sets. ROC curves are calculated for the fixed interval-scale analysis using the testing sets only.

4.1.1. Classification and regression trees (CART)—Using the second framework (Fig. 4b), a decision tree algorithm (CART, [28]) is used as its performance is more robust to unbalanced data. Table 2 includes the average decision tree model performance on both training and test dataset. As can be observed from the results, the decision tree overfit the training dataset. Thus, to overcome this issue, an ensemble of decision trees (Section 4.1.2) is employed.

Fig. 8 includes the ROC curves for the decision tree model in the fixed interval-scale analysis. Based on this figure, the best performance of the decision tree model yields a sensitivity of 88.96 and specificity of 86.30 for a rate of 0.45.

4.1.2. The ensemble of decision trees—Based on the first framework (Fig. 4a), an ensemble of decision trees model is used to improve performance by reducing overfitting and better handle the unbalanced data set. To combat overfitting, the maximal number of decision splits was set to be equal to the number of observations in the training sample. To ameliorate the unbalanced nature of the data, the RUSBoost algorithm [29] is employed. In this algorithm, an intelligent undersampling technique is used to balance the class distribution, which results in a simple algorithm with faster training times and favorable performance. Table 2 includes the result of the algorithm for beat-scale analysis.

Fig. 8 includes the ROC curve for the ensemble of decision trees algorithm in the fixed interval analysis. Based on this figure, the best result is sensitivity of 91.56 and specificity of 91.97 with rate of 0.4.

4.1.3. SVM—In order to train an SVM to implement the first framework (Fig. 4a), the optimization problem needs to be modified to properly handle the unbalanced data. Consider a soft-margin SVM for binary classification with the following formulation:

$$\min_{\mathbf{w},\, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_i \xi_i$$

$$\text{s.t. } y_i\left(\mathbf{w}^T \mathbf{x}_i + b\right) \geq 1 - \xi_i \;\; \xi_i \geq 0$$

where ($\mathbf{x}_i$, y_i) are sample input/label pair, **w** and *b* are parameters of the separating hyperplane, ξ_i are slack variables and $C > 0$ sets the relative importance of maximizing the margin and minimizing the amount of slack to penalize misclassifying an observation. In order to revise the optimization problem of binary classification to handle unbalanced data, *C* is weighted by class populations such that

$$C_k = C\omega_k,\ \omega_k = \frac{1}{2}\frac{n}{n_k},\ k = 0, 1,$$

where ω_k is the weight of class *k, n* is the total number of observations, and n_k is the number of observations in class *k*. This is indeed a cost-sensitive SVM with the following formulation [24]:

$$\min_{\mathbf{w},\, b}\frac{1}{2}\|\mathbf{w}\|^2 + C_0\sum_{i:\, y_i = 0}\xi_i + C_1\sum_{i:\, y_i = 1}\xi_i$$

$$\text{s.t. } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i\ \ \xi_i \geq 0$$

As mentioned earlier, for a normal PPG beat, *all* the features are expected to be close to zero; while for a poor quality interval, the absolute value of *at least* one of these features is expected to be greater than a threshold. This suggests that a Gaussian kernel is good option, thus an SVM model with Gaussian kernel is considered. Table 2 contains the results of the SVM model in beat-scale analysis.

Fig. 8 includes the ROC curve for the fixed interval-scale performance of SVM. It can be seen that the best result has 93.25% sensitivity and 91.90% specificity for rate of 0.7. Fig. 7 depicts a visual example of the SVM model used for classifying PPG signal quality on three intervals.

4.1.4. Threshold optimization—As it was mentioned earlier (Section 3.1.4) and represented in Fig. 4b, another learning algorithm that is considered here involves determining the optimum threshold for each feature. One approach is an algorithm that chooses a threshold that balances the trade-off between sensitivity and specificity. However, this algorithm is inefficient as “poor quality” can be reflected only in one feature, while the labels are assigned to all six features of the sample. For instance, a jump in positive peaks may only be reflected in normalized positive-to-positive peak jump and normalized beat amplitude jump. Moreover, unbalanced data makes such an optimization method even harder. An optimal algorithm would use the sample labels to extract distinct labels for each feature, and then uses those feature labels to find an optimal threshold. However, as developing such an algorithm is challenging, instead the following method is proposed.

In the proposed threshold optimization algorithm, first the good quality class is undersampled using a non-uniform undersampling method. First, using the training data for a predefined $0 < q < 1$, the *q*-quantile of each feature is calculated, keeping only the samples for which at least one feature is greater than the *q*-quantile of that feature. These samples are used to find a threshold for each feature that balances sensitivity and specificity. Since only about 6% of samples are of poor quality, the 0.94-quantile is chosen for the non-uniform undersampling procedure.

Applying the aforementioned non-uniform undersampling and the threshold optimization algorithm, the results of beat-scaled analysis are included in Table 2.

Fig. 8 also includes the ROC curve for the fixed interval-scale analysis of the proposed threshold optimization algorithm. Based on this figure, the best case has 90.05% sensitivity and 89.48% specificity with a rate of 0.65.

5. Discussion and comparison with other methods

Tables 2 and 3 and Fig. 8 compare the results of all four methods for beat-scale, the best performance of fixed interval-scale, and fixed interval-scale ROC curves, respectively. As summarized in Table 2, the decision tree model and threshold optimization both used the second framework in their learning process, while the ensemble of decision trees and SVM used the first framework. The ensemble of decision trees and threshold optimization are the two algorithms that used undersampling. Both of these methods were also significantly faster to train than those which used the first framework. In comparing model performance on the training and testing datasets, the tree based algorithms overfitted the training data, while SVM and the threshold optimization algorithm have almost the same performance on both datasets.

In addition to Table 2, the effect of uniform and non-uniform undersampling has been tested on SVM and decision tree: non-uniform undersampling used in threshold optimization reduces the performance of both algorithms, while uniform undersampling has no significant effect on SVM (in its cost-sensitive SVM formulation) and a negative effect on decision tree performance. Based on Fig. 8 and Table 3, SVM and the ensembles of decision trees outperform the other two methods in the fixed interval-scale analysis.

Overall, the cost-sensitive SVM with Gaussian kernel outperform the rest, while the proposed threshold optimization is significantly faster.

5.1. Comparison with other methods

An exact comparison of the proposed framework with other state-of-the-art algorithms on the ARDS dataset cannot be achieved, as the algorithms and their attendant procedures are not publicly available, nor are the threshold (or hyperparameter) optimization processes of those methods thoroughly described. Additionally, as previously mentioned many of these algorithms also require an ECG signal as input. This difficulty in comparison is common, as many of the previously proposed PPG signal quality assessment methods did not compare the performance of their SQI algorithm with any other methods [1,2,9–13,15,17, 18,21]. As a result, instead of comparing the proposed algorithm with other approaches on the ARDS dataset, the performance of the algorithm is compared with two other algorithms on the publicly available CapnoBase (CB) database used in those studies [10,14].

In this experiment on dataset there are 57149 samples at the beat-scale (Section 3.3). Similar to the previous experiment, 100 iterations of random subsampling at the patient level is performed to divide the CB dataset into $\frac{2}{3}$ training data and $\frac{1}{3}$ testing data. The cost-sensitive SVM (described in Section 4.1.3) is used for machine learning, as it outperforms other methods on the ARDS dataset. Finally, for SQI calculations, the fixed interval-scale analysis (described in Section 3.4) on windows of length 8 s with the rate 0.9 is performed. Table 4 summarizes the comparison of the proposed framework with the best-case scenario of other algorithms (best-case in Refs. [10,14]: assuming that if beat detection is correctly performed, then quality assessment would be accurate, so the overall performance assumed to be affected only with beat detection). The main reason for achieving higher performance on the CB dataset in comparison to the ARDS dataset is the quality of PPG signal in the CB dataset. In contrast to the ARDS dataset, the CB data is recorded during anesthesia, making contaminated segments of data obvious. As such, signal quality assessment on the ARDS dataset is more challenging.

To analyze the effect of window length in a fixed interval-scale analysis of the quality assessment of the proposed algorithm, the same experiment was performed for various window lengths. Table 5 represents the effect of window length on quality assessment. As can be seen, for short window lengths the proposed framework performs poorly, while for window lengths greater than 4 s it performs reasonably well. This behavior is expected as analyzing a window with more than one beat can be more indicative of the quality of that interval.

The advantages of the proposed morphological features and frameworks are three-fold: 1) applicability to any source of pulsatile physiological signals due to the adaptive nature of the proposed algorithm and the definition of the normalized morphological features; 2) independence from any synchronized reference signal such as ECG, with the only essential inputs being the signal under assessment and its sampling frequency; 3) adaptivity to any dataset. Additionally, while many of the proposed signal quality assessment approaches did not use any machine learning methods [9–16], the proposed framework enables usage of machine learning to better investigate the quality of pulsatile signals. Also, unlike various state-of-the-art frameworks that use ECG signal as an input to their respective algorithms [1,2,20,21], the proposed framework is independent of any synchronized signal. These properties make the proposed framework unique.

6. Conclusion

In this paper, a machine learning framework with a set of morphological features is introduced that is able to measure the quality of any pulsatile physiological signal and detect poor quality segments. Different machine learning algorithms were tested against the ARDS dataset, with cost-sensitive SVM and an ensemble of decision trees outperforms all other. Additionally, the cost-sensitive SVM also achieved better performance in comparison with two state-of-the-art algorithms on a publicly available dataset.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgement

We thank Prof. W. Karlen and his team for sharing (publicly available) the CapnoBase (CB) dataset and the annotations.

Conflicts of interest

This work was partially supported by the National Science Foundation under Grant No. 1722801 and by the National Institutes of Health under Grant NHLBI K01HL136687.

The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; or in the writing of the manuscript.

References

[1]. Li Q, Clifford GD. Signal quality and data fusion for false alarm reduction in the intensive care unit. J Electrocardiol 2012;45(6):596–603. [PubMed: 22960167]

[2]. Silva I, Lee J, Mark RG. Signal quality estimation with multichannel adaptive filtering in intensive care settings. IEEE (Inst Electr Electron Eng) Trans Biomed Eng 2012;59(9): 2476–85.

[3]. Rusch T, Sankar R, Scharf J. Signal processing methods for pulse oximetry. Comput Biol Med 1996;26(2):143–59. [PubMed: 8904288]

[4]. Lee CH, Yoon H-J. Medical big data: promise and challenges. Kidney Res Clin Pract 2017;36(1):3. [PubMed: 28392994]

[5]. Berner ES, Kasiraman RK, Yu F, Ray MN, Houston TK. Data quality in the outpatient setting: impact on clinical decision support systems. In: AMIA annual symposium proceedings, vol. 2005 American Medical Informatics Association; 2005 p. 41.

[6]. Hovenga E, Grain H. Clinical decision support systems: data quality management and governance. Health Inf Gov Digit Environ 2013;193:362.

[7]. Mohktar MS, Sukor JA, Redmond SJ, Basilakis J, Lovell NH. Effect of home telehealth data quality on decision support system performance. Procedia Comput Sci 2015;64:352–9.

[8]. Pflugradt M, Moeller B, Orglmeister R. Opra: a fast on-line signal quality estimator for pulsatile signals. IFAC-PapersOnLine 2015;48(20):459–64.

[9]. Sun X, Yang P, Zhang Y-T. Assessment of photoplethysmogram signal quality using morphology integrated with temporal information approach. In: Engineering in medicine and biology society (EMBC), 2012 annual international conference of the IEEE IEEE; 2012 p. 3456–9.

[10]. Karlen W, Kobayashi K, Ansermino JM, Dumont G. Photoplethysmogram signal quality estimation using repeated Gaussian filters and cross-correlation. Physiol Meas 2012;33(10):1617. [PubMed: 22986287]

[11]. Clarke G, Signal quality analysis in pulse oximetry: modelling and detection of motion artifact, Ottawa-Carleton Institute for Biomedical Engineering.

[12]. Orphanidou C, Bonnici T, Charlton P, Clifton D, Vallance D, Tarassenko L. Signal-quality indices for the electrocardiogram and photoplethysmogram: derivation and applications to wireless monitoring. IEEE J Biomed Health Inf 2015;19(3):832–8.

[13]. Selvaraj N, Mendelson Y, Shelley KH, Silverman DG, Chon KH. Statistical approach for the detection of motion/noise artifacts in photoplethysmogram. In: 2011 annual international conference of the IEEE engineering in medicine and biology society IEEE; 2011 p. 4972–5.

[14]. Papini GB, Fonseca P, Aubert XL, Overeem S, Bergmans JW, Vullings R. Photoplethysmography beat detection and pulse morphology quality assessment for signal reliability estimation. In: 2017 39th annual international conference of the IEEE engineering in medicine and biology society (EMBC) IEEE; 2017 p. 117–20.

[15]. Jang D-G, Kwon UK, Yoon SK, Park C, Ku Y, Noh SW, Kim YH. A simple and robust method for determining the quality of cardiovascular signals using the signal similarity. In: 2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC) IEEE; 2018 p. 478–81.

[16]. Vadrevu S, Manikandan MS, Real-time ppg signal quality assessment system for improving battery life and false alarms, IEEE transactions on circuits and systems II: express briefs.

[17]. Li Q, Clifford G. Dynamic time warping and machine learning for signal quality assessment of pulsatile signals. Physiol Meas 2012;33(9): 1491. [PubMed: 22902950]

[18]. Elgendi M Optimal signal quality index for photoplethysmogram signals. Bioengineering 2016;3(4):21.

[19]. Pereira T, Gadhoumi K, Ma M, Xiuyun L, Xiao R, Colorado RA. A supervised approach to robust photoplethysmography quality assessment, IEEE J Biomed Health Inf. https://www.ncbi.nlm.nih.gov/pubmed/30951482.

[20]. Chen L, Dubrawski A, Wang D, Fiterau M, Guillame-Bert M, Bose E, Kaynar AM, Wallace DJ, Guttendorf J, Clermont G, et al. Using supervised machine learning to classify real alerts and artifact in online multi-signal vital sign monitoring data. Crit Care Med 2016;44(7):e456. [PubMed: 26992068]

[21]. Sukor JA, Redmond S, Lovell N. Signal quality measures for pulse oximetry through waveform morphology analysis. Physiol Meas 2011;32(3):369. [PubMed: 21330696]

[22]. Cover TM, Thomas JA. Elements of information theory. John Wiley & Sons; 2012.

[23]. Akbani R, Kwek S, Japkowicz N. Applying support vector machines to imbalanced datasets. In: European conference on machine learning Springer; 2004 p. 39–50.

[24]. Cao P, Zhao D, Zaiane O. An optimized cost-sensitive svm for imbalanced data learning. In: Pacific-asia conference on knowledge discovery and data mining Springer; 2013 p. 280–92.

[25]. He H, Garcia EA. Learning from imbalanced data. IEEE Trans Knowl Data Eng 2009;21(9):1263–84.

[26]. Krawczyk B Learning from imbalanced data: open challenges and future directions. Prog Artif Intell 2016;5(4):221–32.

[27]. Tang Y, Zhang Y-Q, Chawla NV, Krasser S. Svms modeling for highly imbalanced classification. IEEE Trans Sys Man Cybern Part B (Cybern) 2009;39(1):281–8.

[28]. Breiman L. Classification and regression trees. Routledge; 2017.

[29]. Seiffert C, Khoshgoftaar TM, Van Hulse J, Napolitano A. Rusboost: a hybrid approach to alleviating class imbalance. IEEE Trans Syst Man Cybern A Syst Hum 2010;40(1):185–97.

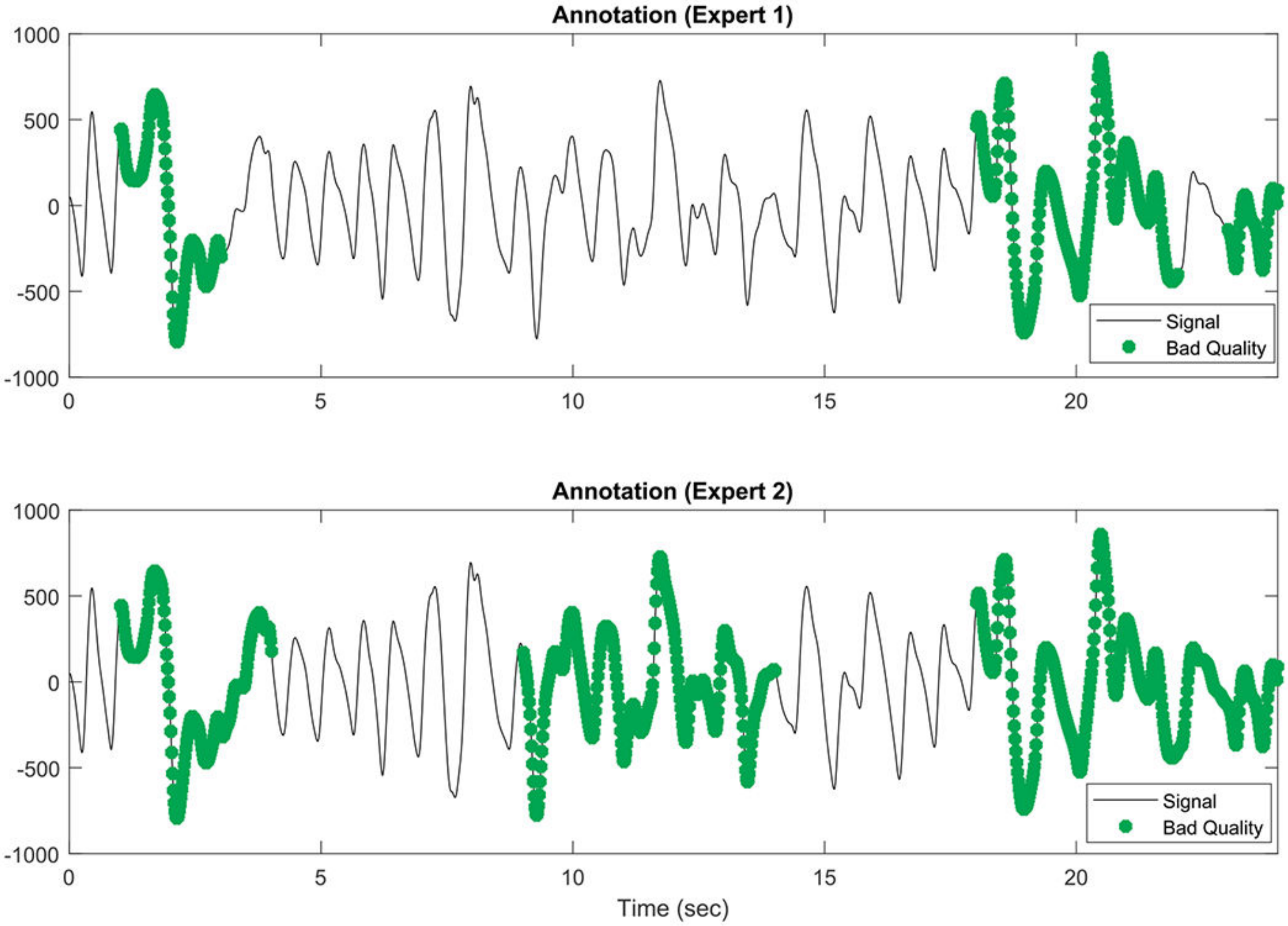

Fig. 1.

Exemplary segment of PPG signals designated with “bad” quality from both experts (clinicians). A signal segment not annotated as bad quality is assumed to be of good quality.

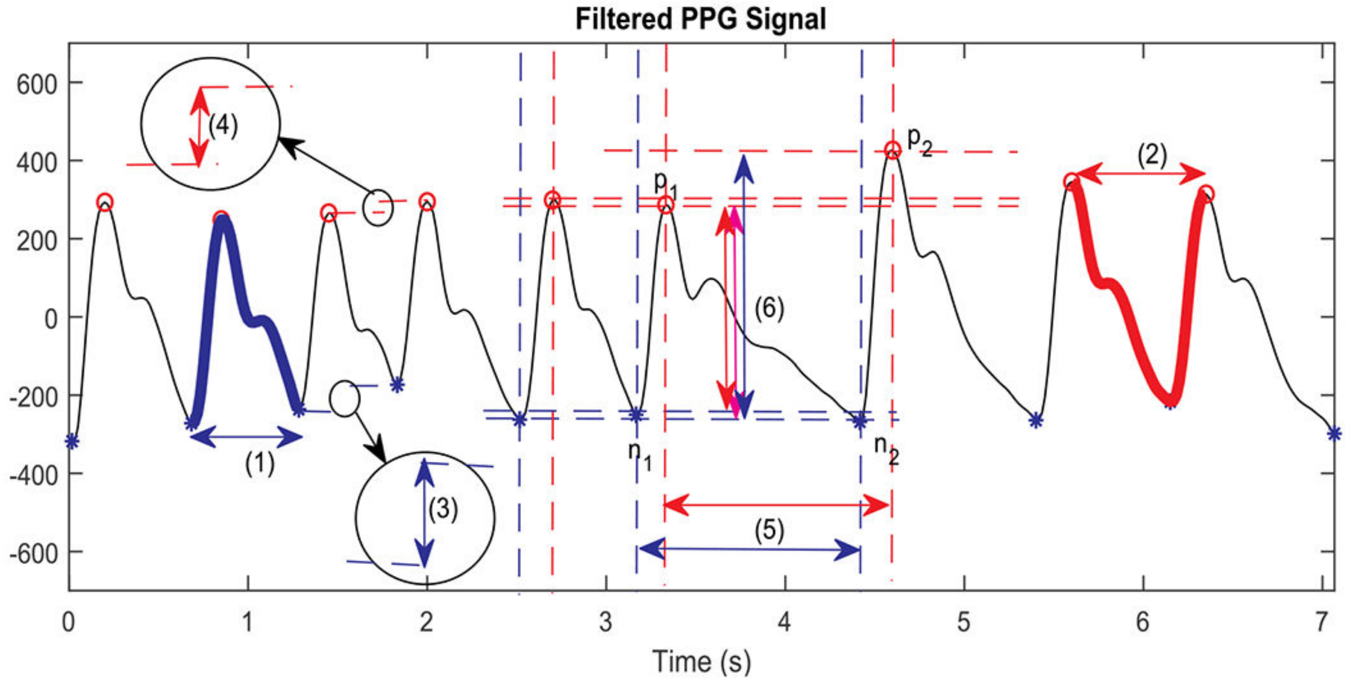

Fig. 2.

The preprocessed (filtering and peak detection) PPG with the following signals/ measurements: (1) beat waveform with positive peak, (2) beat waveform with negative peak, (3) negative-to-negative peak jump, (4) positive-to-positive peak jump, (5) positive and negative pulse duration, and (6) backward and forward AC components.

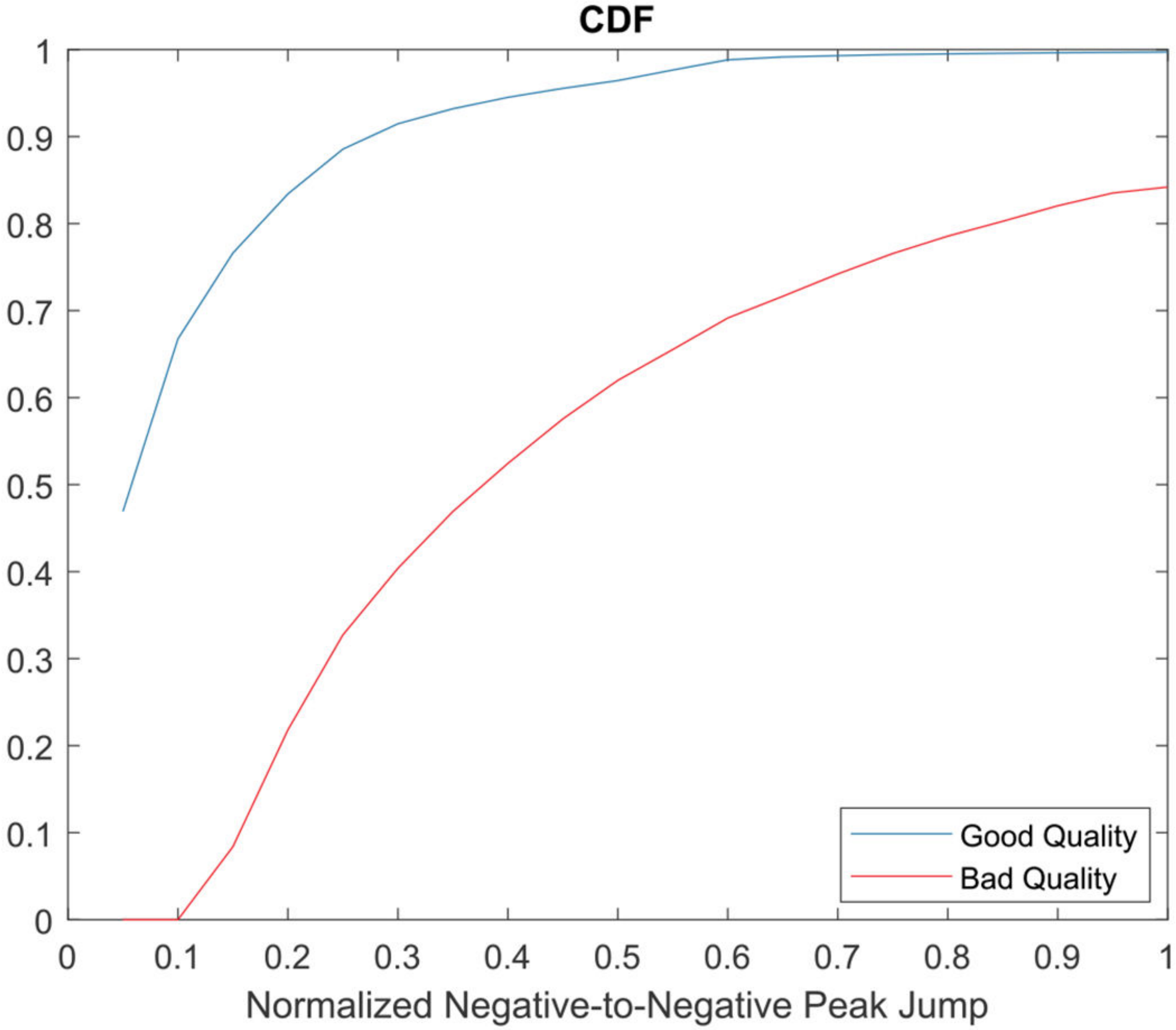

Fig. 3.

Cumulative distribution function (CDF) of normalized negative-to-negative peak jump ($\overline{\nabla P_i^-}$).

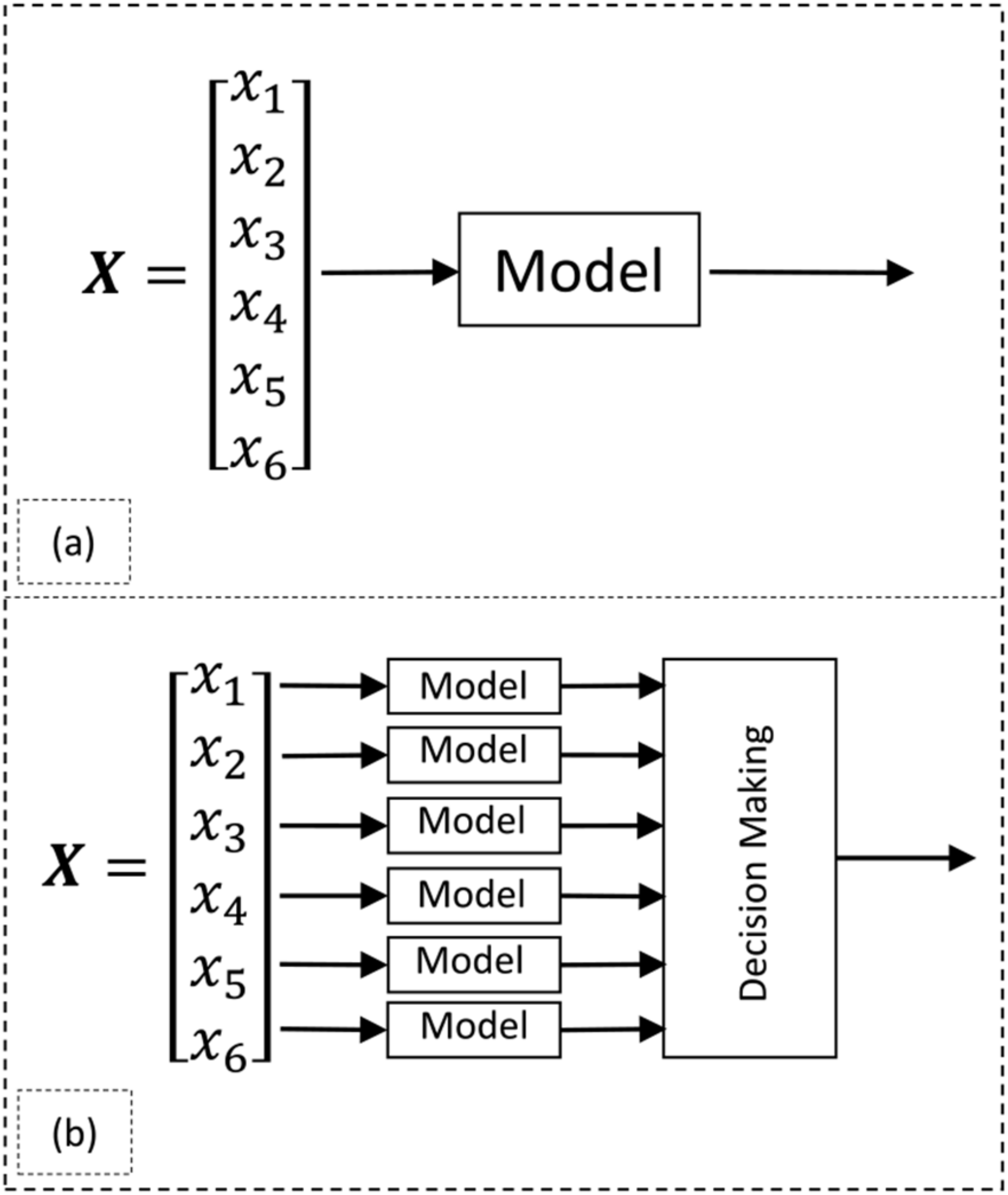
$X = [x_1, x_2, x_3, x_4, x_5, x_6]$
Model
(a)
$X = [x_1, x_2, x_3, x_4, x_5, x_6]$
Model
Model
Model
Model
Model
Model
Decision Making
(b)

Fig. 4.

Two training/testing framework used in this paper: (a) a framework for training/testing model on 6-dimensional samples (b) a framework for training/testing six similar models on each 1-dimensional sample feature followed by decision rule, which basically is a logical “or” operation on the six outcomes.

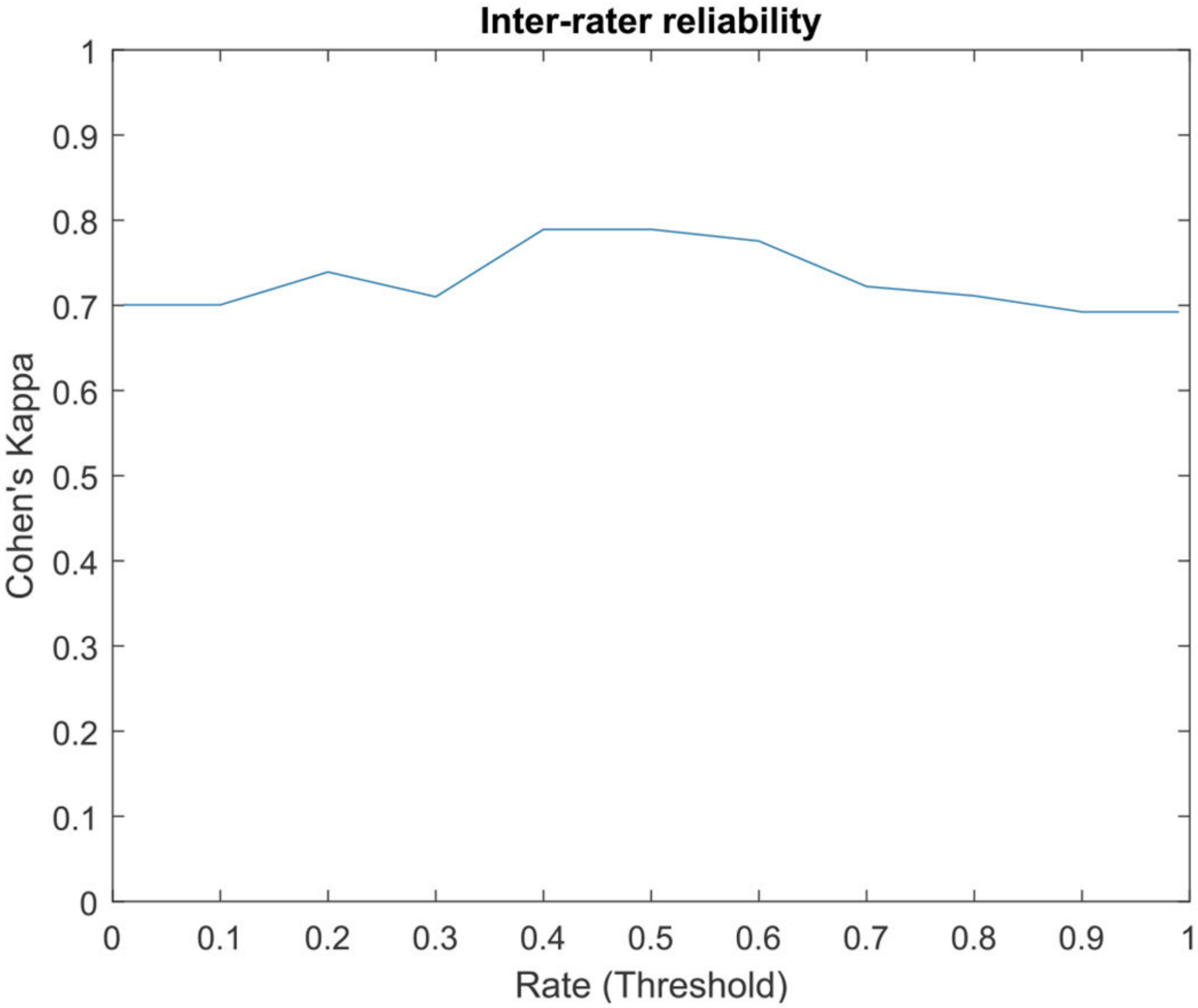

Fig. 5.

Inter-rater reliability using Cohen’s Kappa.

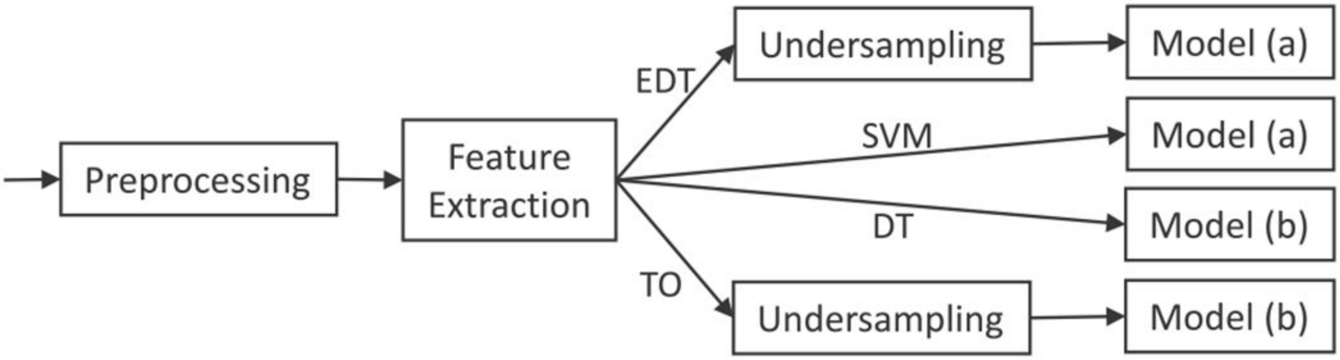

Fig. 6.

A block diagram of learning process. DT: decision tree, EDT: ensemble of decision trees, SVM: support vector machine, TO: threshold optimization. Models (a) and (b) refer to the two training frameworks illustrated in Fig. 4.

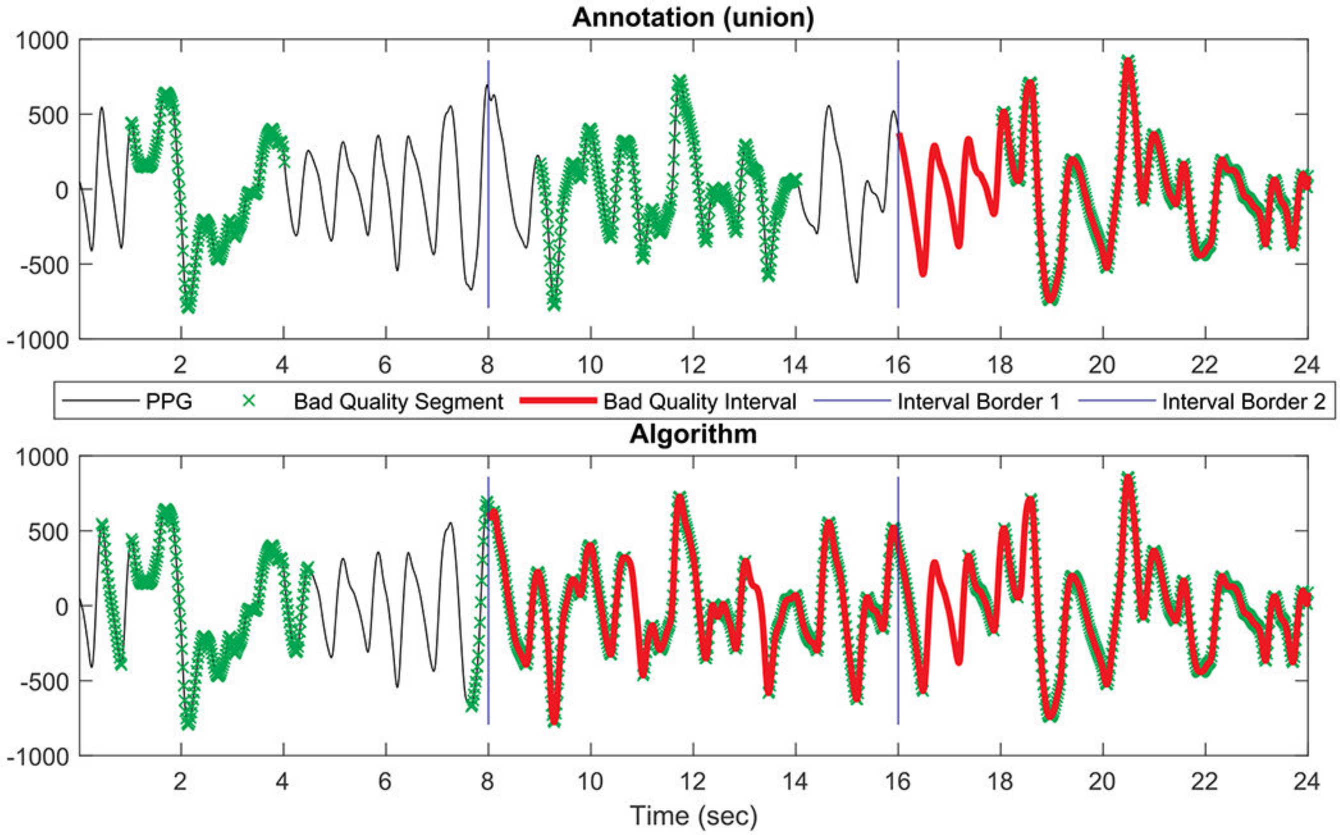

Fig. 7.

A visual example of quality result on fixed interval-scale segments using the SVM model with rate (threshold on interval SQI) 0.7. In the first interval (0–8 s) both the algorithm and annotation have poor quality segments in beat-scale, which is less than 5.6 (8 × 0.7) seconds; thus, this interval is not considered poor quality by both the algorithm and the annotation. The second interval (8–16 s) had more than 5.6 s of poor quality beat-scale segments using the algorithm, but slightly less than 5.6 s of poor quality beat-scale segments using the annotation, therefore this interval is labeled as poor quality using the algorithm, but not using the annotation. The last interval (16–24 s) has more than 5.6 s poor quality beat-scale segments in both algorithm and annotation.

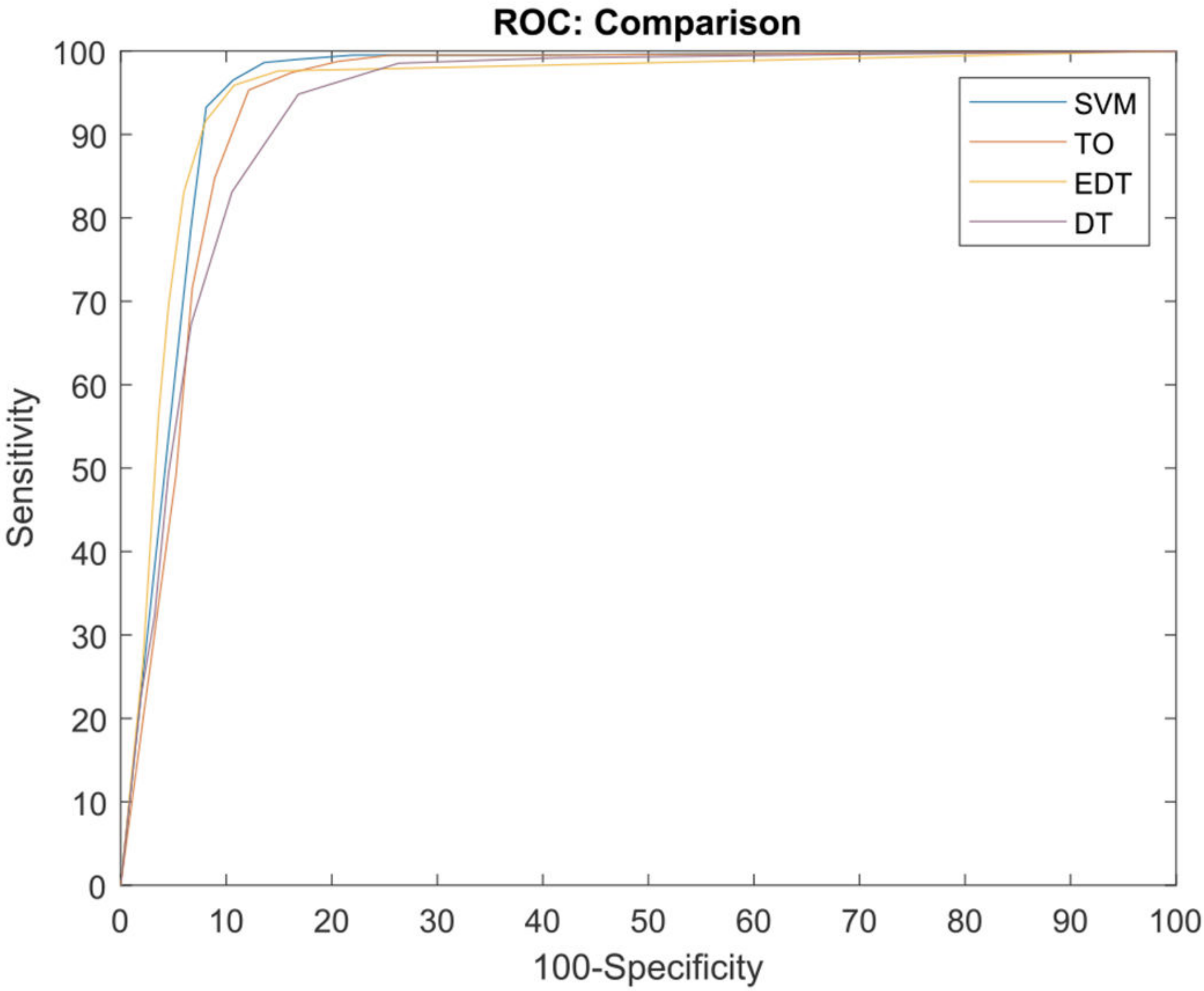

Fig. 8.

Comparison of ROC curves for the four methods used in this study.

Table 1

The unbalanced nature of the ARDS dataset at beat-scale level.

	Expert 1	**Expert 2**	**Union**	**Intersection**
Percentage of poor quality samples	4.46	4.87	6.39	2.94

Table 2

Performance comparison of decision tree (DT), the ensemble of decision trees (EDT), SVM and threshold optimization (TO) in beat-scale analysis. The running time is the average time needed to train the algorithms on 31 30-min PPG signals and test on 15 30-min PPG signals.

		DT	EDT	SVM	TO
Undersampling		No	Yes	No	yes
Framework		Two	One	One	Two
Running time (sec)		5	450	375	20
Train	Accuracy	96.92	100	85.37	80.82
	Sensitivity	99.92	100	86.05	82.82
	Specificity	96.70	100	85.31	80.67
Test	Accuracy	75.02	88.85	83.02	80.66
	Sensitivity	73.01	70.03	85.45	82.38
	Specificity	75.14	90.04	82.82	80.50

Table 3

Comparison of the best performance of decision tree (DT), the ensemble of decision trees (EDT), SVM and threshold optimization (TO) in interval-scale analysis. Please note that the rate in the table corresponds to the best performance.

Best Performance	DT	EDT	SVM	TO
Sensitivity	88.96	91.56	93.25	90.05
Specificity	86.30	91.97	91.90	89.48
Corresponding Rate	0.45	0.4	0.7	0.65

Table 4

Comparison of quality assessment between the proposed algorithm using cost-sensitive SVM and the best-case scenario of the frameworks proposed in Refs. [10,14] on the publicly available Capnobase (CB) dataset.

Best Performance	Proposed Method	[10]	[14]
Sensitivity	98.27	96.44	98.87
PPV	100	99.80	99.22

Table 5

The effect of window length on the prediction of poor quality segments.

Length (Sec)	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Sensitivity	72.77	76.22	82.53	88.39	92.70	96.55	98.27	98.27	98.27	98.27
PPV	82.50	92.68	96.42	98.33	100	100	100	100	100	100