2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

ReLearn: A Robust Machine Learning Framework in Presence of Missing Data for Multimodal Stress Detection from Physiological Signals*

Arman Iranfar, Adriana Arza, and David Atienza[1]

Abstract— **Continuous and multimodal stress detection has been performed recently through wearable devices and machine learning algorithms. However, a well-known and important challenge of working on physiological signals recorded by conventional monitoring devices is missing data due to sensors insufficient contact and interference by other equipment. This challenge becomes more problematic when the user/patient is mentally or physically active or stressed because of more frequent conscious or subconscious movements. In this paper, we propose ReLearn, a robust machine learning framework for stress detection from biomarkers extracted from multimodal physiological signals. ReLearn effectively copes with missing data and outliers both at training and inference phases. ReLearn, composed of machine learning models for feature selection, outlier detection, data imputation, and classification, allows us to classify all samples, including those with missing values at inference. In particular, according to our experiments and stress database, while by discarding all missing data, as a simplistic yet common approach, no prediction can be made for 34% of the data at inference, our approach can achieve accurate predictions, as high as 78%, for missing samples. Also, our experiments show that the proposed framework obtains a cross-validation accuracy of 86.8% even if more than 50% of samples within the features are missing.**

I. INTRODUCTION

Stress, as a global issue of modern societies, increases the risk of several health pathologies, such as, heart diseases, depression, and sleep disorders [1]. Continuous and multimodal stress detection and recognition have been realized through wearable devices and embedded machine-learning algorithms, using stress biomarkers extracted from different physiological signals, such as photoplethysmography (PPG), respiration (RSP), electrodermal activity (EDA), electrocardiogram (ECG), and skin temperature [2]–[5].

A well-known challenge of working on physiological signals recorded by conventional monitoring devices [6] is the presence of missing data due to sensors or electrodes insufficient contact and user's motion, as well as interference by other equipment [7]. This situation deteriorates if the user is physically or mentally active or stressed as a consequence of more frequent conscious or subconscious movements. In general, handling missing data and outliers is of paramount importance when solving real-life classification and regression problems through pattern recognition techniques [8]. On one hand, classification and regression models should be fit offline with a complete and flawless training dataset, i.e., without missing values or outliers. On the other hand, these models, when online, should be still able to provide accurate enough predictions even in presence of missing data and outliers. Therefore, it is vital to have a machine learning framework sufficiently robust to the effect of incomplete data, especially for biomedical applications where prediction accuracy directly or indirectly affects human life quality.

To address such issues in general-purpose applications, researchers have modified the traditional pattern recognition techniques to consider outliers and missing data [9], [10]. However, these techniques lie in certain assumptions, excluding the nature of physiological signal recording and biomarker extraction from wearable devices, where it is possible to encounter a long missing segment of data [11].

Although several works [12], [13] in the biomedical application domain have considered the impact of incomplete data, they are very simplistic, providing inaccurate predictions. Moreover, to the best of our knowledge, stress detection from physiological signals in presence of missing data has not been taken into account in the literature. Therefore, a comprehensive framework that deals with missing data in stress biomarkers throughout its whole pipeline needs to be addressed.

In this work, we propose **ReLearn**, a robust machine learning framework for stress detection from biomarkers extracted from multiple physiological signals that effectively copes with missing data both at training and inference phases. In particular, the training dataset composed of stress biomarkers with missing values flows into a pipeline of feature selection, data imputation, and outlier detection machine learning algorithms, such that the classifier can be fed with a complete training dataset. Then, at inference, unseen data including the missing values are first passed through the trained models obtained from the feature selection, data imputation, and outlier detection algorithms. Finally, the imputed data are used by the classifier to make predictions.

Our main contributions in this work are as follows:

- We propose a novel machine learning framework for multimodal stress detection from physiological signals, which is robust to missing data and outliers.
- According to our experiments and stress database, while by discarding all missing data, as a simplistic yet common approach, no prediction can be made for 34% of the data at inference, our approach is able to achieve accurate predictions, as high as 78%, for missing samples.

*This work has been partially supported by the ML-Edge Swiss National Science Foundation (NSF) Research project (GA No. 200020182009/1), in part by the DeepHealth H2020 Project (GA No. 825111), and by the ONR-G through Award Grant No. N62909-20-1-2063.

[1]A. Iranfar, A. Arza, and D. Atienza are with the Embedded Systems Laboratory of Swiss Federal Institute of Technology Lausanne, Switzerland. {arman.iranfar, adriana.arza, david.atienza}@epfl.ch

- Our experiments show that the proposed framework obtains a cross-validation accuracy of 86.8% even if more than 50% of samples within the features are missing.

II. RELATED WORK

Several works in the literature deal with outliers, noise, and missing data in physiological signals. In this context, [13] only considers parts of signals with valid values, discarding all missing data (NAN values). Nonetheless, throwing away part of the data makes it impossible to have any prediction about the patient's or user's condition. Authors in [12] replace missing values with the closest valid values of the corresponding point. However, such an approach only suits situations where missing data occur infrequently. Also, if more than a few successive data points are missing, filling this gap with the last valid value is insufficient and the outcome could be misleading.

Besides such simplistic approaches, there are several more complex methods addressing missing and noisy data for different applications. In particular, [14] uses a reference channel to substitute the missing data of physiological time series. Since this method works directly on the raw biosignals, it is not suitable for machine learning approaches, where rather than the raw signal, the extracted features are used as the input data. [15] reconstructs the missing leads of a 12-lead ECG signal from a single-lead ECG signal by using the Random Forest algorithm. Nevertheless, this work does not address how to cope with missing data for other physiological signals, such as PPG, RSP, etc. A Singular Value Decomposition (SVD) analysis of outlier detection and imputation of missing data is presented by [16] for DNA microarrays. Similar to [15], the work of [16] is application-specific and cannot be used as a general solution. Adaptive filtering is leveraged by [17] to predict a 30-second segment of missing cardiovascular signals. This approach, however, falls short if long enough segments of signals without any missing data cannot be found before the missing segment. A high dimensional Gaussian Mixture Model (GMM) is built by [8] to address classification problems in high-dimensional samples with missing values. Although this approach shows promising results for surface electromyography (sEMG) signals, it does not provide a holistic solution for multimodal physiological signals. Similarly, authors in [18] propose an adaptive incremental hybrid classifier to alleviate the impact of outliers in myoelectric pattern recognition. A more general-purpose framework is proposed by [19] where an iterative algorithm is used for classifying data with a missing feature. Although the proposed algorithm has been tested on different datasets and applications, it neither considers nor examines the effect of outliers on the classification task.

Our proposed machine learning framework, in contrast to state-of-the-arts, provides a comprehensive solution that can work on arbitrary physiological signals while addressing missing data and outliers.

III. PROPOSED FRAMEWORK

In this work, we design a machine learning framework for stress detection from multimodal physiological signals, which can cope with missing values and outliers at both training and inference time. Fig. 1 shows an overall view of the proposed framework. First, multimodal physiological signals are preprocessed to extract input features (i.e., stress biomarkers) of machine learning algorithms and create the training and testing datasets, both including several samples with missing values. Second, we propose to prune the features, i.e., to exclude those whose ratio of missing value over all samples of the training data is above a particular predefined threshold. So that we reject those features that are prone to missing values, hence, the more affected by the noise and artifacts. However, since important physiological information could be loosed, the trade-off between looser/weaker pruning and cross-validation accuracy is assessed in Section V-A.

Then, the training samples without missing values are used to train the *Data Handler*, where machine learning models for feature selection, outlier detection, and data imputation are trained. Those models are used to clean and impute the training data with missing values, resulting in training samples without any missing values. Thereafter, the data handler is retrained with the complete and enhanced data without missing values to create our ultimate feature selector, data imputer, and outlier detector. Afterward, the machine learning classifier is trained with the training data without any missing values and outliers. At inference time, our ultimate retrained data handler (feature selector, outlier detector, and data imputer) is used to deal with missing values and outliers of unseen testing data prior to the classification. Finally, the trained classifier is employed to provide real-time predictions.

Throughout this framework, we apply cross-validation (CV) with balanced accuracy score [20] for training the feature selection algorithm, as well as the classifier. For this purpose, we apply a group K-fold cross-validation, with K=10, where for each iteration of the cross-validation, a couple of the existing groups (in our case, subjects), are kept for validation, while the training is performed for the rest of the groups. The group K-fold cross-validation is desirable since it can avoid overfitting, particularly, when the samples at inference most likely come from completely different subjects. In what follows, we detail each stage of the proposed framework.

A. Signal Preprocessing

The data-flow from the raw physiological signals to training and testing datasets is shown in Fig. 3. The first step is the signal preprocessing, wherefrom each physiological signal a set of features that capture the subject's physiological stress response is extracted in segmentation widows of 60s.

First, the raw signals are filtered to remove noise and artifacts. Second the filtered signals are delineated to obtain the primary parameters as in [4], [21] and [3], which are shown in Fig. 2. In this step, for each parameter, several

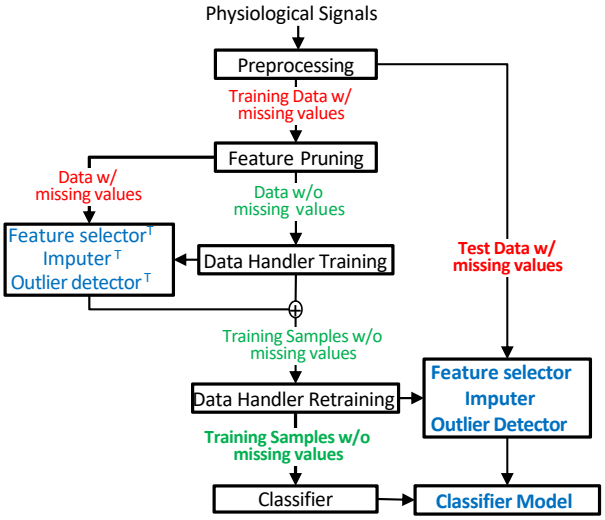

Fig. 1. Overall view of proposed framework for multimodal stress detection in presence of missing data

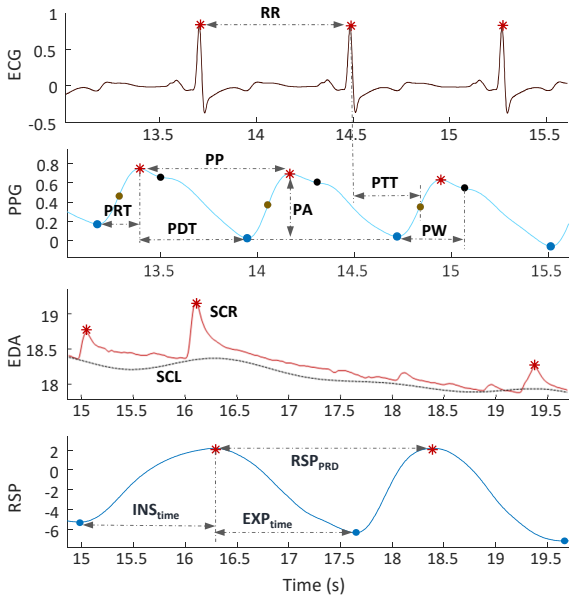

Fig. 2. Biomarkers extracted from electrodermal activity (EDA), respiration (RSP), electrocardiogram (ECG) and photoplethysmography (PPG) signals.

data quality policies are applied mainly based on physiological expected values (e.g., heart rate from 30-180 bpm) and previous samples trend (i.e., within mean, median, and standard deviation values ranges of the last 3 to 5 samples). Besides, each delineation algorithm also rejects noisy signal-segments based on the expected physiological signal shape.

Next, 94 physiological features in the time and frequency domain are extracted from the parameters time series in segmentation windows of 60s, as described in our previous works [3], [22]. Here again, several policies are applied for the missing data on the parameters time series when extracting the features on each segmentation window to ensure the characterization of the physiological response on the sample. For instance, frequency features are only computed if we have more than 98% of the data; heart cycle-based features (on ECG and PPG signals) are valid if more than 10 heartbeats are delineated in a segmentation window; similarly with the respiration cycles, more than 5 cycles. In the case of the EDA signal, its features return a missing value if less than 80% of the data is available. These features are described as follows:

1) EDA: The EDA signal is divided into two main components: Skin Conductance Level (SCL) and Skin Conductance Response (SCR) as the driver phasic signal [23]. Then, the gradient and mean of the SCL, as well as the SCR power are obtained.

2) RSP: We compute respiration period (RSP_{PRD}), duration of air inhaled (INS_{time}), and exhaled (EXP_{time}), and the ratio of inhalation to exhalation duration, from which statistical features are extracted. In the frequency domain, we compute the power and normalized band power of the segmented signal in different frequency bands. Moreover, for each window of analysis, we applied the method proposed in [24] to compute the estimated respiratory frequency, the largest peak power, the total power, and the normalized respiratory peak power.

3) ECG: From ECG, the time intervals between two consecutive R peaks (RR) are obtained. From the RR interval, several time and frequency domain features are extracted based on the Heart Rate Variability (HRV) analysis [25]. Non-linear features are also extracted from the Poincaré plot indicating vagal and sympathetic function.

4) PPG: Several parameters are computed as represented in Fig. 2: pulse period (PP), pulse wave rising time (PRT), pulse wave decreasing time (PDT), pulse width until reflected wave (PW), pulse amplitude (PA), and the slope of the pulse (k) defined as the slope transit time between 1/4 and 3/4 of the PA divided by their difference in amplitude. PP interval features are equal to the aforementioned for RR intervals.

B. Feature Pruning

After feature extraction, we randomly select 70% of the subjects for training, i.e., 66 and 29 subjects' data, respectively, for training and testing, while making sure no subject's data in the training dataset appears in the testing datasets, cf. Fig. 3. We split the data in a stratified fashion such that the same proportion of class labels exists in both datasets.

Our prepossessing stage by applying the missing/noisy data policies results in a dataset with reliable samples but also missing ones. Therefore, we only consider those features of training data that have values for at least more than a predefined percentage of the samples on the training set on the pruning step. The lower the threshold is, the fewer features prone to missing values are included in the training set, hence more reliable information but not necessarily the most relevant one (i.e., most important features). Therefore, our framework includes this percentage threshold as a hyperparameter that needs to be studied and tuned according to the data at hand.

Finally, the training samples are further split into two groups. The first group consists of samples without any

missing value, whereas in the second group, each sample has at least one missing value. Hence, the size of the initial data without any missing value changes with the pruning threshold selection.

C. Data Handler: Feature Selection, Imputation, and Inlier Detection

Fig. 4 illustrates the building blocks of the proposed data handler that aims at creating a dataset without any outliers and missing values while including only significant features. The training samples without missing values obtained from the preprocessing stage are used in this stage of our framework to build the baseline models for feature selection, data imputation, and inlier detection to be, then, applied to the training samples with missing values.

The first step is to exclude outliers in the training data. We use Isolation Forest [26], an anomaly detection algorithm, to find the outliers. After removing the outliers, we use Recursive Feature Elimination with Cross-Validation (RFECV) to automatically find the most significant features. We utilize the Random Forest algorithm as the baseline classifier of the RFECV algorithm. After training RFECV, a feature selection model is obtained which can be later used to eliminate unnecessary features of the wider input feature sets. Using the Isolation Forest prior to RFECV provides a more robust model for feature selection since RFECV can work better on clean data, without outliers.

After selecting the most relevant features, we use a multivariate iterative imputer [27], where missing values are imputed by modeling each feature with missing values as a function of other features in an iterated round-robin fashion [28]. We employ the Bayesian Ridge algorithm [29], [30] as the baseline regressor of the multivariate imputer. This step provides us with an imputer model, which is later used to impute the missing data. Since the training dataset in this step does not have any missing values, this step achieves a reliable imputer, even with a limited number of samples.

Having known the most relevant features, we propose to retrain the Isolation Forest model obtained in the first step of the data handler. The reason lies in the fact that we intend to use our inlier detector model on a selection of the input features. We propose to refit our inlier detection model after applying the imputer, since in a real scenario it is first required to impute the missing values, otherwise, the outlier detector fails to find the true outliers.

Having obtained the models of feature selection, feature imputation, and inlier detection, we pass the second part of the training data containing missing values through these models, achieving training samples without unnecessary features, outliers, and missing values (v2). This already-cleaned part of training data can then be concatenated to the first part of the training data (v1) to create the final training dataset (v3).

Although using the initial clean and complete training data, v1, in the data handler provides us with the models of feature selection, imputation, and inlier detection, these models have been trained on a subset of the training dataset, i.e., those samples initially without any missing values. If this part of the training data is not sufficiently large, the overall framework is prone to overfitting. Therefore, one solution is to retrain the feature selector, multivariate imputer, and inlier detector with the complete and larger training dataset (v3) obtained from the *initial* data handler. As a consequence of retraining the machine learning models within the data handler, new models for feature selection, inlier detection, and data imputation are attained, which can be later used at inference time. We refer to these models as *retrained* data handler models.

D. Classifier

To find the best classifier, we perform a grid search with cross-validation for several classical machine learning algorithms, including Linear Discriminant Analysis, Support Vector Classifier, Random Forest, and eXtreme Gradient Boosting (XGB) classifier similar to [31]. Although we only consider five of the well-known machine learning classifiers, our approach is not limited to incorporating these classifiers and any arbitrary machine learning algorithm can be employed within our proposed framework. In the grid search, we consider the most important hyperparameters of these algorithms. We found XGB able to provide a statistically higher CV score than the others. Therefore, we use XGB as the main classifier of the proposed framework.

IV. EXPERIMENTAL SETUP

To assess our proposed technique, we build our framework in Scikit-Learn [32]. Then, we test it with experimental data from [33] on a single core of a 32 AMD EPYC Processor with a maximum frequency of 2 GHz, a 500 GB main memory, and an 8 MB Last Level Cache (LLC).

A. Stress Database: Experiment Protocol

95 participants (male, $Age_{mean} = 20.43$, $Age_{std} = 2.17$) are divided into two groups performing either a control or a stress task in a virtual reality (VR) environment lasting 10 minutes each one [33]. The physiological signals are recorded using the Biopac BioNomadix System. The stress experiment is approved by the Cantonal Ethics Committee of Vaud, Switzerland (2017-00449). The stress task exposed participants to an uncontrollable social-evaluative task and timed problem solving with negative feedback in a challenge in VR. Here, participants were immersed in an empty room with tiled flooring, in which they could move around while mental arithmetic questions appeared briefly in the heads-up display (HUD). Incorrect responses caused a tile on the floor to break and disappear, leaving an open hole where participants could fall into. Performance was continuously compared to a faux average performance from other participants (63% of correct responses; being also shown in the HUD) and the difficulty (response time limit) was titrated to keep performance below this average. The control task consisted of equivalent conditions but without the stressful elements of the stress task. Participants were still standing up and allowed to walk while being immersed in a VR nature setting.

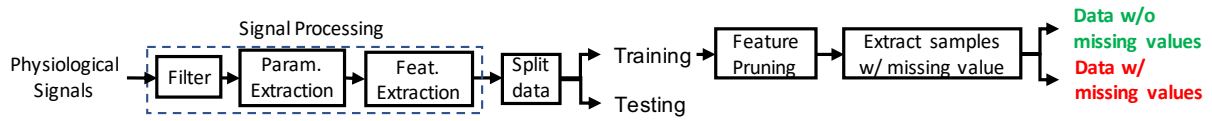

Fig. 3. Preprocessing stage of our proposed framework

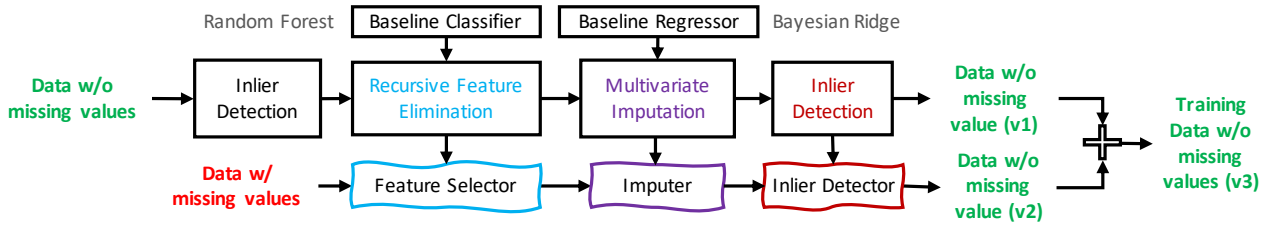

Fig. 4. Data handler of proposed framework

B. Evaluation of the Proposed Framework

To evaluate the utility of our framework to enhance the training data we compare the use of the data handler trained only by data without missing values against the two-step process of retraining our data handler on the complete training data that includes both the initial clean and treated data with the first data handler, see Fig. 1. In particular, we compare how it behaves with respect to the different thresholds of the feature pruning, hence how robust our data handler is when the training data have more missing values.

Moreover, we compare our framework with traditional techniques for handling missing data and outliers. The most conventional techniques for replacing the missing values are 1) to fill the gaps with the mean value of valid samples (i.e., those with neither missing values nor outliers) computed from the training set and 2) to replace the missing value with the value of the last valid sample. In the first data imputation technique, the mean value of each feature obtained from the training dataset is used to fill the missing values. In addition, we also consider in our comparison the very basic yet commonly used alternative to handle missing values, where all the missing values are simply ignored [13], discarding entire rows containing missing values. However, this comes at the price of losing valuable data. Finally, the most common, yet simplistic, approach to consider the outliers is removing any values lying in a distance beyond 3 times of standard deviation from the mean value. We also assess this technique.

The same signal processing as the one explained in Section III-A is applied to the data prior to using these traditional techniques. Also, to have a fair comparison, we apply RFECV for feature selection followed by performing a grid search over the hyperparameters of the classifier (XGB) with cross-validation. Finally, we remove any features that more than 50% of their values in the training dataset are missing.

V. EXPERIMENTAL RESULTS AND DISCUSSION

A. Feature Pruning Threshold Selection

One of the steps taken in our framework in the preprocessing stage is to prune the features, i.e., to discard those features that more than a particular percentage of their samples in the training set are missing.

Fig. 5 shows how this threshold affects the mean and standard deviation of CV accuracy. As shown in the figure, the CV mean accuracy of all threshold values range from 83.0% to 86.8% (solid line with markers) with a standard deviation from 6.4% to 11.2%, indicating the robustness of the proposed framework. It is important to note that these results are highly dependent on the dataset used since the affected features with missing values vary with the sensor used and activities performed.

Moreover, the choice of threshold value affects the number of selected features and overhead of the framework. TABLE I shows the number of features after feature pruning, training samples without missing values after feature pruning used in the initial data handler, number of features after the retrained data handler used for training the classifier, and overhead of the whole framework at inference. As shown by TABLE I, the execution time proportionally increases with the number of features. In fact, the greater the number of features selected, the more complex would be the model of the imputer, inlier detector, and the final classifier.

We argue that either of the thresholds evaluated in this work can be considered as the chosen value for the pruning step, depending on the user's intent and the application targeted. On one hand, if the main purpose is to attain a low-overhead solution, a threshold value of 10% to 20% is a proper choice as it can achieve an acceptable score of 85.15% with a minimum runtime overhead of only 0.8 ms. On the other hand, a threshold of 50% provides slightly higher accuracy, i.e., 86.83% at the cost of larger runtime overhead ($1.8ms$). The standard deviation of accuracy in cross-validation is, nonetheless, quite large for smaller thresholds. In this work, we assume 50% as the threshold of missing data in the feature pruning step, as it provides the highest CV mean accuracy and the lowest standard deviation.

B. Framework Analysis and Evaluation

As explained in Section III-C the models extracted from the initial data handler models can also be used at inference time. However, as aforementioned, the initial data handler is trained on only a part of the training data, v1. Therefore, the final feature selector, data imputer, and outlier detector

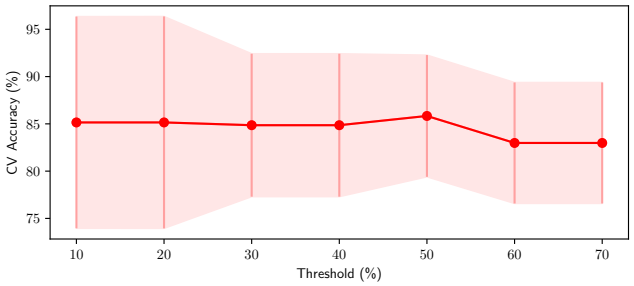

Fig. 5. Impact of threshold for feature pruning cross validation results

TABLE I

IMPACT OF DIFFERENT THRESHOLDS FOR FEATURE PRUNING

Threshold (%)	10	20	30	40	50	60
#Features (After Feat. Prun.)	66	66	80	80	86	87
#Samples w/o missing values (After Feat. Prun.)	2475	2475	1925	1925	1498	675
#Features (Final)	21	21	27	27	38	30
Overhead (ms)	0.8	0.8	1.5	1.5	1.8	1.6

from the retrained data handler using the whole clean dataset may outperform the models provided by the initial data handler. We test this hypothesis by deploying the proposed framework with the initial and retrained data handlers at different threshold values for feature pruning.

Fig. 6 shows the box plots of CV score achieved by models of the initial data handler and retrained data handler for different threshold values. With lower thresholds, the initial data handler attains a higher mean CV score than retrained data handler with a larger standard deviation. By increasing the threshold, retrained data handler provides not only a higher CV mean score, but also a lower standard deviation. As a consequence, inferring from the models of the retrained data handler brings about more robustness against missing data. Also, if applied to unseen testing data, with a feature pruning threshold of 50%, the classification mean accuracy achieved through the retrained and initial data handlers are 78.8% ($std = 25.4\%$) and 76.5% ($std = 26.4$), respectively. Moreover, Fig. 7 depicts the number of features selected up to each of the approaches and the overall runtime overhead of the framework. According to this figure, the models of retrained data handler consistently use fewer features and, hence, come with lower complexity with different threshold values. Therefore, inferring from the models of retrained data handler results in reduced complexity while having statistically more accurate predictions.

C. Comparison with Conventional Imputation Techniques

In this section, we evaluate our approach against traditional techniques for coping with missing data and outliers. For this comparison, we assume the following approaches: filling the missing values with a mean value *Mean value*; the last value*Last Value* and ignoring all the missing values *Drop NAN*, on the training dataset.

TABLE II compares our approach and these techniques with respect to the CV mean accuracy and standard deviation, inference mean accuracy and standard deviation among subjects on the unseen test data, mean accuracy of predictions

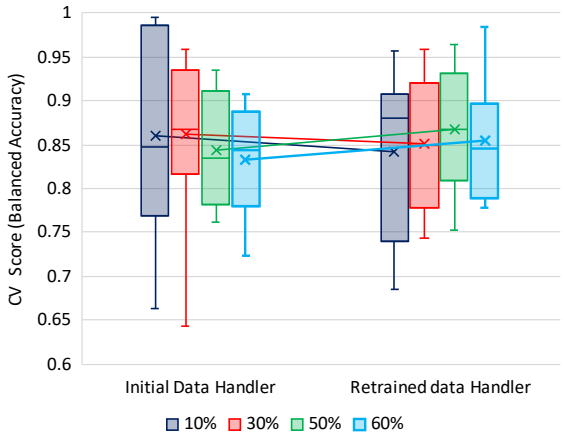

Fig. 6. CV-accuracy obtained from models of the initial and retrained data handlers with respect to different threshold values

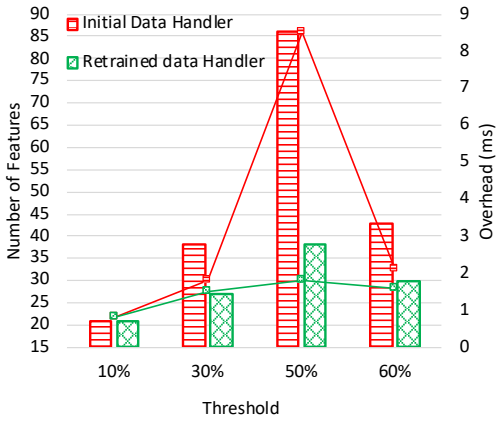

Fig. 7. Number of features (bars) and overhead obtained by the initial and retrained data handlers with respect to different threshold values

on missing data at inference, and the number of selected features. As shown by TABLE II, by ignoring all missing data, the CV mean accuracy is close to that obtained by our framework, yet 4% less. Besides, this technique results in a rather large standard deviation (12.0%), thus, lower accuracy at inference. More importantly, regarding our database, the *Drop NAN* method fails to provide any prediction for 387 samples out of 1131 samples. This is more than 34% of the data to be predicted at inference. Even if we let only the features with less than 30% of their values missing (instead of 50% used in TABLE II), still no prediction can be made for 16% of samples for unseen testing data. In this case, the CV mean accuracy and inference accuracy increase to 84% and 77.4%, yet lower than the one achieved by the proposed framework for all threshold values discussed in Section V-A.

In contrast to the *Drop NAN* method, using *Mean Value* and *Last value* techniques reduce overfitting through data imputation and, thus, increasing the training data, at the cost of lower CV and inference accuracy. In addition, despite the fact that predictions on missing data are made available by these two techniques, neither of them can reach the same accuracy as that provided by the proposed framework. The poor classification accuracy obtained through these two methods

TABLE II

COMPARISON TO CONVENTIONAL IMPUTATION TECHNIQUES

	CV (mean ± STD)	Inf. (All) (mean ± STD)	Inf. (Miss.)	#Feat.
Drop NaN	81.6 ± 12.0%	74.4 ± 30.0%	N/A	24
Mean value	53.9 ± 13.5%	50.2 ± 38.7%	51.1%	49
Last Value	54.1 ± 12.4%	49.7 ± 38.2%	50.5%	36
ReLearn	**86.8 ± 6.4%**	**78.8 ± 25.4%**	**77.9%**	**38**

is mainly due to the large number of missing samples in the training and testing datasets, which necessitates a more complicated solution rather than these simple imputation techniques.

VI. CONCLUSION

In this paper, we have proposed ReLearn, a new robust machine learning framework for stress detection from biomarkers extracted from multimodal physiological signals that effectively addresses missing data and outliers both at training and inference phases. Our framework enables efficiently increasing and cleaning the training data. Thus, it provides a more accurate and generalizable classification. Moreover, our framework allows classifying all samples at inference, including missing ones. In particular, according to our experiments in a large stress database, while by discarding all missing data as a simplistic yet common approach, no prediction can be made for 34% of the data at inference, our approach is able to achieve very accurate predictions, as high as 78%, for missing samples. Furthermore, we have shown that our approach facilitates the use of features that are usually discarded due to missing values, despite containing significant information of the physiological stress response. Thus, our approach achieves a cross-validation and inference accuracy of 86.8% and 78.8%, respectively, even if up to 50% of samples within the features are missing.

REFERENCES

[1] S. Cohen, D. Janicki-Deverts, and G. E. Miller, "Psychological stress and disease," *Jama*, vol. 298, no. 14, pp. 1685–1687, 2007.

[2] E. Smets, W. De Raedt, and C. Van Hoof, "Into the Wild: The Challenges of Physiological Stress Detection in Laboratory and Ambulatory Settings," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 2, pp. 463–473, Nov. 2019.

[3] V. Montesinos Canovas, *et al.*, "Multi-modal acute stress recognition using off-the-shelf wearable devices," in *International Engineering in Medicine and Biology Conference*. IEEE, July 2019, pp. 2196–2201.

[4] A. Arza, *et al.*, "Measuring acute stress response through physiological signals: towards a quantitative assessment of stress," *Medical & biological engineering & computing*, vol. 57, no. 1, pp. 271–287, 2019.

[5] M. Parent, *et al.*, "PASS: A Multimodal Database of Physical Activity and Stress for Mobile Passive Body/ Brain-Computer Interface Research," *Frontiers in Neuroscience*, vol. 14, p. 542934, 2020.

[6] X. D. Zhang, Z. Zhang, and D. Wang, "Cgmanalyzer: an r package for analyzing continuous glucose monitoring studies," *Bioinformatics*, vol. 34, no. 9, pp. 1609–1611, 2018.

[7] X. Dong, *et al.*, "An improved method of handling missing values in the analysis of sample entropy for continuous monitoring of physiological signals," *Entropy*, vol. 21, no. 3, 2019.

[8] Q. Ding, *et al.*, "Missing-data classification with the extended full-dimensional gaussian mixture model: Applications to emg-based motion recognition," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 8, pp. 4994–5005, 2015.

[9] P. Melchior and A. D. Goulding, "Filling the gaps: Gaussian mixture models from noisy, truncated or incomplete samples," *Astronomy and Computing*, vol. 25, pp. 183 – 194, Oct. 2018.

[10] T. Chen, E. Martin, and G. Montague, "Robust probabilistic pca with missing data and contribution analysis for outlier detection," *Computational Statistics & Data Analysis*, vol. 53, no. 10, pp. 3706–3716, 2009.

[11] G. B. Moody, "The physionet/computing in cardiology challenge 2010: Mind the gap," in *2010 Computing in Cardiology*. IEEE, 2010, pp. 305–308.

[12] S. M. Jadhav, S. Nalbalwar, and A. Ghatol, "Artificial neural network based cardiac arrhythmia classification using ecg signal data," in *2010 International Conference on Electronics and Information Engineering*, vol. 1. IEEE, 2010, pp. V1–228.

[13] Y. Bai, Y. Guan, and W.-F. Ng, "Fatigue assessment using ecg and actigraphy sensors," in *Proceedings of the 2020 International Symposium on Wearable Computers*, 2020, pp. 12–16.

[14] P. Langley, *et al.*, "Estimation of missing data in multi-channel physiological time-series by average substitution with timing from a reference channel," in *2010 Computing in Cardiology*. IEEE, 2010, pp. 309–312.

[15] K. Afrin, *et al.*, "Simultaneous 12-lead electrocardiogram synthesis using a single-lead ecg signal: Application to handheld ecg devices," *arXiv preprint arXiv:1811.08035*, 2018.

[16] L. Liu, *et al.*, "Robust singular value decomposition analysis of microarray data," *Proceedings of the National Academy of Sciences*, vol. 100, no. 23, pp. 13 167–13 172, 2003.

[17] A. Hartmann, "Reconstruction of missing cardiovascular signals using adaptive filtering," in *2010 Computing in Cardiology*. IEEE, 2010, pp. 321–324.

[18] Q. Ding, *et al.*, "Adaptive hybrid classifier for myoelectric pattern recognition against the interferences of outlier motion, muscle fatigue, and electrode doffing," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 5, pp. 1071–1080, 2019.

[19] M. A. Kiasari, G.-J. Jang, and M. Lee, "Novel iterative approach using generative and discriminative models for classification with missing features," *Neurocomputing*, vol. 225, pp. 23–30, 2017.

[20] K. H. Brodersen, *et al.*, "The balanced accuracy and its posterior distribution," in *2010 20th international conference on pattern recognition*. IEEE, 2010, pp. 3121–3124.

[21] F. Dell'Agnola, *et al.*, "Cognitive workload monitoring in virtual reality based rescue missions with drones," in *12th International Conference on Virtual, Augmented and Mixed Reality*, 7 2020.

[22] G. Masinelli, *et al.*, "Self-aware machine learning for multimodal workload monitoring during manual labor on edge wearable sensors," *IEEE Design and Test*, vol. 2356, pp. 1–7, 2020.

[23] M. Benedek and C. Kaernbach, "A continuous measure of phasic electrodermal activity," *Journal of Neuroscience Methods*, vol. 190, no. 1, pp. 80–91, 6 2010.

[24] A. Hernando, *et al.*, "Inclusion of Respiratory Frequency Information in Heart Rate Variability Analysis for Stress Assessment," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 4, pp. 1016–1025, 2016.

[25] Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology., "Heart rate variability: standards of measurement, physiological interpretation and clinical use." *Circulation*, vol. 93, no. 5, pp. 1043–1065, 3 1996.

[26] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *International Conference on Data Mining*. IEEE, 2008, pp. 413–422.

[27] S. v. Buuren and K. Groothuis-Oudshoorn, "mice: Multivariate imputation by chained equations in r," *Journal of statistical software*, pp. 1–68, 2010.

[28] F. Pedregosa, *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[29] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *Journal of machine learning research*, vol. 1, no. Jun, pp. 211–244, 2001.

[30] D. J. MacKay, "Bayesian interpolation," *Neural computation*, vol. 4, no. 3, pp. 415–447, 1992.

[31] N. Momeni, *et al.*, "Real-time cognitive workload monitoring based on machine learning using physiological signals in rescue missions," in *International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2019, pp. 3779–3785.

[32] F. Pedregosa, *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[33] J. Rodrigues, *et al.*, "Locomotion in virtual environments predicts cardiovascular responsiveness to subsequent stressful challenges," *Nature Communications*, vol. 11, no. 1, p. 5904, 2020.