Hindawi
Complexity
Volume 2018, Article ID 8651930, 13 pages
https://doi.org/10.1155/2018/8651930

WILEY | Hindawi

Research Article

Tensor Decomposition for Multiple-Instance Classification of High-Order Medical Data

Thomas Papastergiou, Evangelia I. Zacharaki, and Vasileios Megalooikonomou

Computer Engineering and Informatics Department, University of Patras, Rio, Achaia 26504, Greece

Correspondence should be addressed to Thomas Papastergiou; papastergiou@ceid.upatras.gr

Received 1 June 2018; Accepted 30 August 2018; Published 6 December 2018

Academic Editor: Panayiotis Vlamos

Multidimensional data that occur in a variety of applications in clinical diagnostics and health care can naturally be represented by multidimensional arrays (i.e., tensors). Tensor decompositions offer valuable and powerful tools for latent concept discovery that can handle effectively missing values and noise. We propose a seamless, application-independent feature extraction and multiple-instance (MI) classification method, which represents the raw multidimensional, possibly incomplete, data by means of learning a high-order dictionary. The effectiveness of the proposed method is demonstrated in two application scenarios: (i) prediction of frailty in older people using multisensor recordings and (ii) breast cancer classification based on histopathology images. The proposed method outperforms or is comparable to the state-of-the-art multiple-instance learning classifiers highlighting its potential for computer-assisted diagnosis and health care support.

1. Introduction

Nowadays, data tend to be large in volume and multiparametric in nature, especially in clinical diagnostics and health care. Applications that provide massive multidimensional data are vast. Some examples include monitoring patients by multisensor technologies [1, 2], noninvasive lesion detection and diagnosis using hyperspectral sampling [3], cancer diagnosis based on tissue microarray data [4, 5], color segmentation of skin lesions using histology-stained microscopic images [4, 5], classification of EEG signals for seizure detection [6], or for Alzheimer's disease analysis [7]. The main challenge is to extract discriminative features from high-dimensional data in a way that preserves their multidimensional structure while at the same time models the interdimensions' interaction. Traditional matrix representation techniques that represent high-dimensional data by flattening them to a matrix suffer many times from the curse of dimensionality that poses limitations on many two-dimensional approaches. By representing such data in a more natural way by multidimensional arrays (a.k.a. tensors) and using sophisticated high-order techniques, such as tensor decompositions, we can capture multiple interactions and couplings and simultaneously discover latent concepts that are present in the data [8]. Tensor-based techniques have been employed in the field of signal processing and machine learning for a variety of tasks [9] like in blind multiuser code-division multiple access (CDMA) communications, blind source separation, collaborative filtering-based recommender systems, Gaussian mixture parameter estimation, topic modeling, or, as mostly related to our work, multilinear discriminative subspace learning [10, 11], among many others. For an extensive overview of the underlying tensor theory and the aforementioned applications, we refer to the extensive review paper [9]. Tensor decomposition has also been applied recently for image restoration by grouping image patches [12] or for image compression and reconstruction [13, 14] by removing redundancy simultaneously in spatial and spectral domain. In contrast to multichannel signal or image data encoding that often benefits from tensor decomposition due to their structured nature, encoding of 3D geometrical meshes rather relies on traditional techniques, such as graph Fourier Transform [15]. A common aspect in most of the applications is the exploitation of sparsity in high-order structures. An overview

of some basic techniques that exploit sparsity in the recovery of low-rank higher-order tensors, followed by related applications, is provided in [16].

The second challenge in the analysis of current biomedical data comes in the learning phase that follows the data representation phase. Standard supervised learning implies that each example used for training a classification model, is represented as a feature vector with an associate class label attached. However, in many real-life applications, data tend to be complex, incorporating different concepts, and thus it is difficult to model each example as a single feature vector: e.g., medical images depicting different tissue types, biosignals tracking different activities, or molecules with conformations with different chemical properties. In these cases, a more efficient representation, which preserves as much information as possible, consists of a collection of feature vectors (denoted as instances), such as patches of an image, time windows of biosignals, or conformations of a molecule, each one covering a different aspect of the whole object. The challenge that arises for such representations is the lack of refined annotation for each individual feature vector, known as *multiple-instance learning* (MIL). Furthermore, some of the feature vectors describing an observation could provide none or sometimes even misleading information about the object's class (e.g., not all cells are malignant in a histopathology image with malignancy).

Besides the challenges inherited by the high-order structure and multivariate context, data partiality or incompleteness impose an additional burden. Missing data occur in real-life due to a variety of reasons including failure in the data acquisition processes (e.g., temporary malfunction of EEG electrodes [17]), costly experiments impeding the annotation of all samples, or due to noise or artifacts removal. In supervised learning paradigms, missing values must be removed from the data or imputed by statistical approaches [18] prior to inference. Another interesting approach when classifying data with missing values is based on the assumption that data are of low rank [19, 20], that there exist prototypes (i.e., components) and all the samples can be reconstructed by a mixture of them. For example in [19], the classification problem is treated as a matrix completion problem via rank minimization, while in [20], classification is performed using the low-rank assumption without any matrix completion step. For high-dimensional settings, dissimilarity-based classification is proposed in [21] where missing values are estimated via high-order decomposition and then classification is performed on the completed data.

The aim of this work is to define a generalized tensor-based multiple-instance learning framework (called TensMIL) for analyzing high-order, possibly incomplete data, avoiding the extraction of predefined or hand-crafted features. Our approach is formulated as a multistep minimization problem in which all parameters, internal and external, are learnt by supervision. In order to illustrate the wide applicability of TensMIL, we assess it in two distinct scenarios for multiple-instance classification using biomedical images and multichannel biosignals, respectively, and compare it against other state-of-the-art techniques. In order to place the method into the MIL context and better appreciate its differences from other approaches, we first provide a small overview of the related work in MIL and then proceed with more details and contributions of TensMIL.

In multiple-instance learning problems, bags (subjects) are described by multiple-feature arrays (instances) and labels are provided only for the bags, whereas the labels of the individual instances are unknown. Several methods have been proposed exploiting local or global information and implementing different classifiers or mapping functions. For a complete taxonomy on MIL algorithms, we refer to the work of Amores [22], as well as previous reviews by Foulds and Frank [23] or Dong [24]. At the first level of the taxonomy tree, the classification frameworks follow either the Instance Space (IS), Bag Space (BS), or Embedded Space (ES) paradigm.

The inference process for the methods in the IS paradigm is based on information that resides in the individual instances, i.e., an instance-level classifier is trained to separate the instances in positive or negative class. The obtained instance-level scores are then aggregated to summarize the information about the whole bag, usually based on one of the two assumptions [22, 23]: the *standard MI* assumption that states that every positive bag contains *at least one* positive instance and the *collective* (or *weighted collective*) assumption in which all instances in a bag contribute equally (or according to weights) to the bag's label [25]. The selected aggregation rule thus acts as a bag-level classifier. Although the assumption-based IS paradigm proves to be an effective heuristic in many application domains, very often, the relationship between instances in a bag and the bag-level class labels is unknown; therefore, the use of *concepts* was introduced to relax the strict view of predefined assumptions. A more refined hierarchy of assumptions was defined by Weidmann et al. [26] and presented by increasing generality from the standard MI (for a single concept), to the presence-based (for multiple concepts), threshold-based, and count-based assumption.

In contrast to the IS paradigm, where the (bag-level) classifier is obtained as an aggregation of local responses, the inference process of the methods in the BS and ES paradigms is performed in the space of bags. BS methods directly employ a distance or kernel function that operates on non-vectorial entities, such as the bags, in order to assess similarity between them. Since our proposed method relates less to this category of methods, we omit further discussion, but refer to [22] for additional details. In the ES paradigm, a set of concepts are identified by unsupervised learning and used as a vocabulary that describes classes of instances. A mapping function is then employed to map each bag into a feature vector $\mathbf{v}$ which aggregates the pertinent information about the bag. In the special case of histogram-based ES methods, the vector $\mathbf{v}$ describes the distribution (histogram) of the instances into the different classes of the vocabulary. The few ES methods that are not based on vocabularies or concepts' learning usually summarize statistics (for example, the minimum and maximum values) of the features of all the instances inside the bag. Another interesting approach is associating the bags with their most informative instances via instance selection. In this way, the bag space is mapped

to a reduced instance space, where IS classifiers or even classic non-MIL classifiers can be exploited. Recently a new multiple-instance learning algorithm with discriminative bag mapping (MILDM) [27] has been proposed, where informative instances are selected such that the bags are maximally distinguishable in the new mapping space.

In this paper, we propose a seamless method for feature extraction and MIL classification of high-dimensional data by modeling the data as n-dimensional arrays (i.e., tensors). Through tensor decomposition, we construct a high-dimensional dictionary that models the latent factors of the data as a number of $n-1$ dimensional rank-1 constructs. In this way, the coefficients that correspond to the instances' mode indicate the contribution of each latent factor to the representation of the corresponding instance, and thus they can serve as instance-level features. Subsequently, using these features, we train an instance-level classifier for predicting the hidden class label of each instance by a continuous score. We model each bag by the density function of the predicted labels and train a bag space classifier for the final classification task. Our motivation was to avoid strict predefined MIL rules, such as the standard MIL assumption; therefore, we extended the collective assumption, by learning the bag labels using the probability density function of the estimated hidden instances' labels.

The main contributions of our work are summarized as follows:

(1) TensMIL is based on a generalized feature extraction method for high-dimensional data using tensor decomposition, thus can be applied in multiple scenarios

(2) It performs well even with a very small number (e.g., 10%) of observed data

(3) Evaluation in the UCSB Breast Cancer benchmark dataset with full and with partial observed values showed that it outperforms or is comparable to existing state-of-the-art MIL algorithms

(4) To the best of our knowledge, we are the first to exploit the potential of physiological (such as respiratory and cardiac) signals in predicting aging-associated decline (frailty). The application of TensMIL revealed prognostic capabilities for frailty manifestation that previous methods failed to uncover

2. Materials and Methods

The proposed methodology is illustrated in the simplified schematic diagram in Figure 1 and consists mainly of three phases: (i) the data representation and feature extraction phase in which the data are mapped from the original high-dimensional space to a lower dimensional space using tensor decomposition, (ii) the multiple-instance learning phase in which sequential discriminative models are inferred to classify the data into different groups, and (iii) the optimization phase that is coupled with the previous phase for learning the hyperparameters. In the next sections, we describe

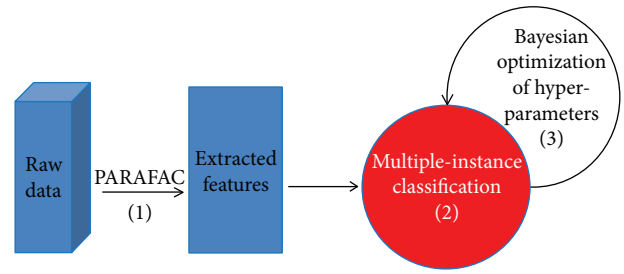

Figure 1: Schematic diagram of the proposed methodology.

analytically every phase starting from the use of tensor decomposition for feature extraction and proceeding with our proposed MIL framework.

The notation that we follow within this manuscript is as follows. We denote tensors by capital boldface Euler letters ($\boldsymbol{\mathcal{X}}$, $\boldsymbol{\mathcal{Y}}$, $\boldsymbol{\mathcal{Z}}$), matrices by capital boldface letters ($\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$), vectors by boldface lowercase letters ($\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$), and scalars by lowercase letters (a, b, c). Entries of a matrix or a tensor are denoted by lowercase letters with subscripts (e.g., the $(i_1, i_2, \ldots, i_n)$ entry of an n-way tensor $\boldsymbol{\mathcal{X}}$ is denoted by $x_{i_1,i_2,\ldots,i_n}$). Columns of a matrix are denoted by a boldface capital letter with a subscript consisting of a star and a number (e.g., $\mathbf{A}_{*,1}$ denotes the first column of matrix $\mathbf{A}$).

2.1. Tensor Decomposition. We briefly outline the CANDECOMP/PARAFAC (CP) decomposition, a powerful tool originally introduced in [28, 29]. For preliminaries on tensors, we refer to the Supplementary Material (available here). Without loss of generality and for the sake of simplicity from now and on, we will refer to 3rd order tensors, although the proposed method can be generalized for high-order tensors. Let $\boldsymbol{\mathcal{X}}$ be a 3-way tensor of size $I \times J \times K$. With full data, a tensor $\boldsymbol{\mathcal{X}}$ can be decomposed into a set of matrices $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$ of sizes $I \times R$, $J \times R$, $K \times R$, respectively, as follows:

$$\boldsymbol{\mathcal{X}} \approx \sum_{r=1}^{R} \mathbf{U}_{*,r} \circ \mathbf{V}_{*,r} \circ \mathbf{W}_{*,r}, \tag{1}$$

where R is the rank of the decomposition and "$\circ$" denotes the outer product of two arrays.

Let Ω be the set of the observed indices of tensor $\boldsymbol{\mathcal{X}}$. We can define an indicator tensor $\boldsymbol{\mathcal{W}}$ having the same size as the original tensor such that $\boldsymbol{\mathcal{W}}(i, j, k) = 1$, $\forall (i, j, k) \in \Omega$, and zero elsewhere. The tensor decomposition problem can then be formulated as follows:

$$\min_{\mathbf{U},\mathbf{V},\mathbf{W}} \left\| \boldsymbol{\mathcal{W}} \circledast \left(\boldsymbol{\mathcal{X}} - \sum_{r=1}^{R} \mathbf{U}_{*,r} \circ \mathbf{V}_{*,r} \circ \mathbf{W}_{*,r} \right) \right\|_{\mathrm{F}}^{2}, \tag{2}$$

where the "$\circledast$" denotes the Hadamard (element-wise) product. When Ω is equal to the set of indices of $\boldsymbol{\mathcal{X}}$, then we have a full- (nonmissing) value decomposition problem; otherwise, we have a decomposition problem with missing values.

For calculating the CP decomposition, we exploit the well-known Alternating Least Squares (ALS) method [30] when we deal with a full-value problem, and the two Proximal methods proposed in [31] when we deal with missing value problems. The methods proposed in [31]—GenProxSGD (nondistributed) and StrProxSGD (distributed, suitable for big data)—tackle the optimization problem in (2) by solving local minimization problems rather than solving the entire problem at once.

2.2. Generalized Feature Extraction. We propose here a general method for extracting instance-based features from raw data in which data are represented as an n-dimensional tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$. The representation of the data is problem-specific, and we will discuss in a later section the representation of data for the two different problems that we tackle. Our objective is to calculate the latent factors of data via the CP decomposition of the raw data tensor, where instances are arranged in one dimension. The obtained factor matrix (the one corresponding to the instances) can be used as feature matrix in the instances' space. The other factor matrices correspond to the calculated high-order dictionary.

Formally, if $\mathcal{X} \in R^{I \times J \times K}$(instances are arranged across the first dimension), we can write slice-wise a rank-R CP decomposition of $\mathcal{X}$ presented in (1) as

$$\mathcal{X}_{i,*,*} \approx \sum_{r=1}^{R} u_{ir}(\boldsymbol{V}_{*r} \circ \boldsymbol{W}_{*r}), \tag{3}$$

where $\mathcal{X}_{i,*,*}$ represents a mode-1 slice of the tensor that corresponds to the ith instance. Equation (3) denotes that each instance can be approximated as a linear combination of R two-dimensional components, $\boldsymbol{V}_{*,r} \circ \boldsymbol{W}_{*,r} \in \mathbb{R}^{J \times K}$ which correspond to the latent factors of the data. Thus, we can choose as features representing an instance i, the R coefficients u_{ir}, $r = 1, 2, \ldots, R$, that correspond to the ith row of factor matrix $\boldsymbol{U}$ in (2). Furthermore, we can see the latent factors $\boldsymbol{V}_{*,r} \circ \boldsymbol{W}_{*,r}$ as a high-order dictionary describing the data. This procedure can be employed as is to tensors of order $N > 3$ yielding dictionaries of order $N - 1$ and is independent of the nature of the data per se.

2.3. Alternative Feature Extraction for New (Unseen) Data. The tensor-based feature extraction process in the proposed framework involves the decomposition of a common tensor constructed by the concatenation of training and testing samples, as described above. For reducing the computational cost, it might be desired to classify new testing data without repeating the whole tensor decomposition. We describe next an alternative approach to obtaining the low-dimensional feature representation in which a PARAFAC model is constructed only from the training data while the test data are represented by the estimated training model as follows. If $\mathcal{X}_{\text{train}} \approx \sum_{r=1}^{R} \mathbf{U}_{*,r} \circ \mathbf{V}_{*,r} \circ \mathbf{W}_{*,r}$ is a PARAFAC decomposition of rank R calculated for the training set and $\mathcal{X}_{\text{test}}$ is the tensor of test data, then it can be shown [30] that the PARAFAC calculation problem of (2) can be written in a mode-1 matricized form as

$$\min_{U,V,W} \left\| \mathbf{X}_{\text{train}(1)} - \mathbf{U}(\mathbf{W} \odot \mathbf{V})^{\mathbf{T}} \right\|_{\text{Fr}}^{2}. \tag{4}$$

We can formulate and solve a least squares minimization problem to find the "closest" representation of the test set based on the calculated dictionary of $\mathbf{U}$ and $\mathbf{W}$:

$$\min_{\tilde{U}} \left\| \mathbf{X}_{\text{test}(1)} - \tilde{\mathbf{U}}(\mathbf{W} \odot \mathbf{V})^{\mathbf{T}} \right\|_{\text{Fr}}^{2}. \tag{5}$$

It is easy to show [30] that the solution of the problem of (5) has the following closed form $\tilde{\boldsymbol{U}} = \boldsymbol{X}_{test(1)}(\boldsymbol{W} \odot \boldsymbol{V})(\boldsymbol{W}^{\boldsymbol{T}}\boldsymbol{W} \circledast \boldsymbol{V}^{\boldsymbol{T}}\boldsymbol{V})^{\dagger}$, where "†" is the Moore-Penrose pseudoinverse.

In the following, we describe the next phase of the methodology that involves the construction of the discriminative model by multiple-instance learning.

2.4. Problem Statement in Multiple-Instance Learning (MIL). We first briefly define formally the multiple-instance learning problem. A bag $B_i = \{\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, \ldots, \boldsymbol{x}_{i,n_i}\}$ is a set of n_i feature vectors describing a subject. Let us denote $\mathbf{B} = \{B_i, i = 1, 2, \ldots, n\}$ as the set of all the bags. The cardinality of each bag B_i can vary across the bags. Each feature vector $\boldsymbol{x}_{i,j}$, where the first index refers to the corresponding bag and the second index to the feature of the bag it belongs to, is called an *instance*. All instances $\boldsymbol{x}_{i,j}$, $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, n_i$ live in a d-dimensional feature space ($\boldsymbol{x}_{i,j} \in \mathbb{R}^d$), called *instance space*. Each bag comes with a label attached to it $Y_i \in \mathcal{Y} = \{1, \ldots, C\}$, $i = 1, 2, \ldots, n$, with $C = 2$ defining a binary classification problem and $C > 2$ defining a C class classification problem. $\mathcal{Y}$ denotes the set of all bag class labels.

The objective of a MIL problem is given a collection of n bags (subjects) with their appropriate labels $\{(B_i, Y_i), i = 1, 2, \ldots, n\}$ to learn a model that can predict the labels of new observations (bags).

2.5. Our MIL Framework (TensMIL). The proposed MIL framework follows the IS paradigm in which an instance-level classifier $f(\boldsymbol{x})$ is first constructed based on the label inheritance rule (i.e., all instances of a bag inherit the label of the bag). In order to make learning computationally feasible, it is generally necessary to reduce the hypothesis space by enforcing some MI assumption. However, in contrast to the classical IS-based methods that directly combine the instance-level responses through some predefined rule, we increase the generality and try to infer those assumptions based on the training set. Specifically, we extract the histogram of all instance-level responses within each bag and learn the distribution of those histograms from the training set. The instance-label responses refer to the output of the instance-level classifier $f(\mathbf{x})$ and are analogous to class prediction scores for each instance. The histogram extraction of the instance-level

responses corresponds to quantizing the responses within predefined bins that can be considered as clusters of low, medium, or high class-likeness. In that sense, our framework relates also to the *ES methods without vocabularies* with the difference that the representation is not based on the original (multiple) attributes of the instances, but on the instance-level responses (output of the first classifier). Our contribution lies in the fact that we do not rely on a few statistics, like the average, minimum, or maximum values, but incorporate a richer representation such as the histogram.

In mathematical terms, we formulate (similarly to previous work [32]) an optimization problem that we solve based on the following steps:

(i) First, the instance-level responses within each bag are estimated based on a function $f(\bullet|\boldsymbol{\theta}_f)$ that assigns a class prediction score (such as an abnormality score) to each instance in the bag given a set of parameters $\boldsymbol{\theta}_f$ (6), by initializing the unknown instance labels with the corresponding class label, i.e., $y_{i,j} = Y_i, \forall_i$:

$$\widehat{\boldsymbol{\theta}}_f = \arg\ \min_{\theta_f} \sum_{i=1}^{n} \sum_{j=1}^{n_i} l\left(f\left(\boldsymbol{x}_{i,j}|\boldsymbol{\theta}_f\right), y_{i,j}\right), \quad (6)$$

where $l : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$ is a loss function defined over the instance space. Upon estimation of $\widehat{\boldsymbol{\theta}}_f$, the function f will provide the predictions for the instance-level class labels, which is in contrast to the work in [32], where the unknown instance-level class labels are considered as optimization variables and are calculated in an iterative manner

(ii) Then, a mapping function $\mathscr{H}(\bullet|\theta_H)$ is applied from the instance space to the bag space and the mapped features are used as the new bag representation $\tilde{B}$ (7). In the proposed method, this mapping corresponds to the calculation of the density function of the class prediction scores and is obtained by histogram extraction:

$$\tilde{B}_i = \left\{\mathscr{H}\left(f\left(\mathbf{x}_{i,j}\middle|\widehat{\boldsymbol{\theta}}_f\right)|\theta_H\right) : \mathbf{x}_{i,j} \in B_i\right\}. \quad (7)$$

(iii) Finally, the classification function $F(\bullet|\boldsymbol{\theta}_F)$ for the whole bag is calculated by supervised learning as shown in the following equations:

$$\widehat{\boldsymbol{\theta}}_F = \arg\ \min_{\boldsymbol{\theta}_F} \sum_{i=1}^{n} L\left(F\left(\tilde{B}_i|\boldsymbol{\theta}_F\right), \mathrm{Y}_i\right), \quad (8)$$

$$\widehat{\mathrm{Y}}_i = F\left(\tilde{B}_i\middle|\widehat{\theta}_F\right), \quad (9)$$

where $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}^+$ is a loss function defined over the bag space

More details on the individual steps are provided in the following sections.

2.5.1. Robust Estimation of the Instances' Hidden Labels. The medical applications usually concern classification problems of ordinal data, where the classes have a natural order, such as the grade of a tumor or the performance score in a clinical test. If class labels are used, they can be considered as a discrete approximation of the continuous score (e.g., malignancy); thus, the same techniques can be applied for discrete or continuous output variables. The binary classification is a special case of this problem, where the two classes lie on the two extremes (minimum and maximum) of the clinical score range.

In the first step (6), we use the squared error as loss function and train a full quadratic regression model (containing an intercept, linear terms, interactions, and squared terms) $f : \mathbb{R}^d \to \mathbb{R}$ in the instance space that predicts the hidden class labels $y_{i,j}$ for each instance. The quadratic regression model can be expressed as

$$f(\boldsymbol{x}) = \sum_{k=1}^{d} \sum_{m=1}^{d} \theta_{km} x_k x_m + \sum_{k=1}^{d} x_k, \quad (10)$$

where the parameters θ_{km} collectively form the vector $\boldsymbol{\theta}_f$ in (6), and d is the dimensionality of $\boldsymbol{x}$ employed in the regression. Since there is no available information about the instances' hidden class labels, the regression model is trained by using values for the dependent variable, the class labels of the corresponding bags, this means that $y_{i,j} = Y_i, \forall j = 1, \ldots, n_i$. Upon the calculation of f, which is common for all bags, we can estimate the instance labels as $\widehat{y}_{i,j} = f(\boldsymbol{x}_{i,j})$.

Since not all the instances of a bag i will belong to the bag's class Y_i, some of the instances will behave as outliers and will not fit well to the respective class. To eliminate the effect of such inconsistent data, we employ robust quadratic regression which uses iteratively reweighted least squares with a weighting function [33]. We used the logistic weighting function:

$$w_k = \frac{\tan h(r_k)}{r_k}, \quad r_k = \frac{\mathrm{resid}_k}{\left(\mathrm{tune}^* s^* \sqrt{1 - h_k}\right)} k = 1, 2, \ldots, d, \quad (11)$$

where **resid** is the vector of residuals of the previous iteration, s is an estimate of the standard deviation of the error term given by the median absolute deviation of the residuals from their mean scaled by a constant z, $\boldsymbol{h}$ is the vector of the leverage values from least-squared fit, and *tune* is a tuning parameter. For the experiments of this paper, we used the default values for the aforementioned parameters: $z = 0.6745$ and $tune = 1.205$. The choice of the constant z makes the estimate of the standard deviation of the error term unbiased for normal distributions. Furthermore, the choice of the above default values gives coefficient estimates that are approximately 95% as statistically efficient as the

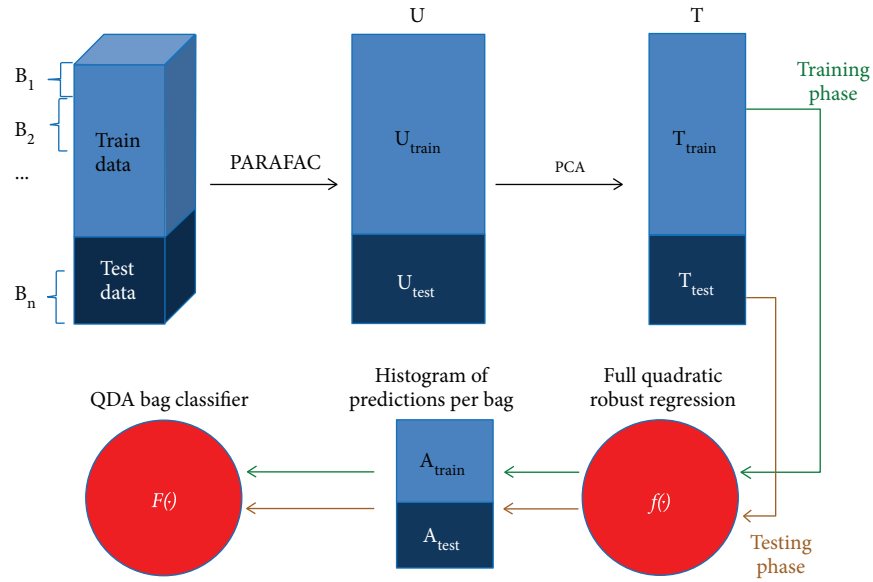

Figure 2: The architecture of TensMIL, where U is the feature matrix extracted from the raw data by PARAFAC decomposition, T is the score matrix obtained by performing PCA on U, A is the matrix containing the bag-level features, $f(\cdot)$ is the full quadratic regression model, and $F(\cdot)$ is the QDA classifier.

ordinary least squares estimates, provided that the response has a normal distribution with no outliers. By employing the above weighting function, the misclassification penalty for the instances that do not belong to the bag's class is reduced, obtaining thus a robust estimation of the hidden labels of the instances. Finally, we want to mention that we experimented with different weighting functions and different tuning parameters and we empirically concluded to use the aforementioned logistic weighting function with the default tuning settings since it yielded better results.

2.5.2. QDA-Based Bag Classification. In order to obtain the bag representation (7) and subsequent bag classification ((8) and (9)), we treat the extracted attributes in target bags (i.e., the instance-level class predictions per bag) as random variables that are defined over a space of probability distributions. We then approximate the density functions $\mathscr{H}_i(\{f(\boldsymbol{x}_{i,j}), j = 1, 2, \ldots, n_i\}), i = 1, 2, \ldots, n$, of the class label scores for each bag by histogram extraction using θ_H equally sized bins. Having estimated the histograms for all bags in the training set $\mathbf{H} = \{\mathscr{H}_i, i = 1, 2, \ldots, n\}$, we can train a bag-wise classifier that will learn to discriminate the unknown class Y. Assuming that the observations from each class $k, k = 1, 2, \ldots, C$ are drawn from a multivariate Gaussian distribution $\mathscr{H} \sim N(\mu_k, \boldsymbol{\Sigma}_k)$ and that each class has its own covariance matrix $(\boldsymbol{\Sigma}_k)$, we can use the quadratic discriminant analysis (QDA) classifier [34] to find a nonlinear quadratic decision boundary. The QDA classifier $F : \tilde{\boldsymbol{B}} \to \mathscr{Y}$ assigns an observation to the class with the maximum discriminant score $\hat{Y}_i = \text{argmax}_k\, \delta_k(h_i)$:

$$\delta_k(p) = -\frac{1}{2}(p - \mu_k)^T \boldsymbol{\Sigma}_k^{\dagger}(p - \mu_k) + \log \pi_k - \frac{1}{2} \log |\boldsymbol{\Sigma}_k|, \quad (12)$$

where δ_k is the discriminant function over the bag space, μ_k is the mean vector of all the training observations from the kth class, $\boldsymbol{\Sigma}_k$ is the covariance matrix for the kth class, and π_k is the prior probability of an observation belonging to the k th class. The parameters $(\mu_k, \boldsymbol{\Sigma}_k)$ of the discriminant functions are learnt from the training set and subsequently used in the testing phase to predict the class labels for new bags.

2.6. Implementation Details and Summary of TensMIL Architecture. In this section, we summarize the individual steps of the method, starting from the raw multidimensional data, and illustrate them in Figure 2 highlighting the differences between training and testing phase.

In the first phase, data must be arranged in a tensor of order $N \geq 3$, with the first dimension dedicated to the instances. The tensor can be constructed by placing instances of each bag $B_1, B_2, \ldots, B_n$ in a sequential order, but this is only for convenience. Training and test data can be placed in the same tensor, constructing a high-dimensional tensor as can be seen in Figure 2. In the second phase (the feature extraction phase), a PARAFAC model is computed and the train and test features are extracted from the corresponding rows of the factor matrix corresponding to the instances' dimension. In the third step, the train and test feature matrices are concatenated along the dimension corresponding to instances and PCA is performed for decorrelation and dimensionality reduction obtaining truncated train and test matrices. The percentage (θ_p) of variance explained in the PCA loading matrix is a parameter of the method and can vary for different datasets. In the fourth step, a robust quadratic regression model is trained for predicting the instances' labels. Finally, the histograms of the class predictions of each bag are then calculated and fitted to a pseudoquadratic discriminant analysis classifier.

Input: training and test instances' features $\mathbf{U}_{\mathbf{train}}$ and $\mathbf{U}_{\mathbf{test}}$, subjects' training labels $\mathbf{Y}_{\mathbf{train}}$, percentage of variance retained by PCA θ_p, the number of bins used for the histograms (θ_H)
Output: prediction model
1. Concatenate $\mathbf{U}_{\mathbf{train}}$ and $\mathbf{U}_{\mathbf{test}}$ along the first dimension into a matrix **U.**
2. Perform PCA for decorrelation and dimensionality reduction on the concatenated matrix **U** and get the scores **T**, using the m-leading singular values that preserve θ_p of data variance.
3. Split the truncated scores matrix **T** into the corresponding $\mathbf{T}_{\mathbf{train}}$ and $\mathbf{T}_{\mathbf{test}}$ (will be used in the testing phase) scores matrix.
4. Train a robust full quadratic regression model (Equation (10)) using $\mathbf{T}_{\mathbf{train}}$ and $\mathbf{y}_{\mathbf{train}}$ (the instance labels inherited by the corresponding bag labels) and get the instance labels predictions $\mathbf{Pred}_{\mathbf{train}}$ for each instance
5. Split the vector $\mathbf{Pred}_{\mathbf{train}}$ into θ_H subsets of equal sizes and store the cutting points to be used as histogram bin edges in the testing phase
6. For each of the n training bags calculate the normalized cumulative histogram and construct the $n \times \theta_H$ feature matrix $\mathbf{A}_{\mathbf{train}}$
7. Fit a QDA model F to map $\mathbf{A}_{\mathbf{train}}$ to $\mathbf{Y}_{\mathbf{train}}$ (Equation (12)).

Algorithm 1: TensMIL (training)

2.6.1. Bayesian Optimization of Hyperparameters. The parameters of the two incorporated models, $\boldsymbol{\theta}_{\mathbf{f}}$ and $\boldsymbol{\theta}_{\mathbf{F}}$, are calculated sequentially by supervised learning, whereas the number of histogram bins (θ_H) and the percentage (θ_p) of variance retained from the set of hyperparameters are optimized externally and used as input in the learning phase. We optimized the hyperparameters using Bayesian optimization [35], based on 2-fold cross-validation on the training set.

The algorithm for the training phase of TensMIL is shown in Algorithm 1.

2.7. Assessment of the Method. As evaluation metrics for the selection of the hyperparameters and overall assessment of the methodology, we used the classification accuracy (number of correctly classified samples over total number of samples), the balanced accuracy, and the area under the ROC curve (AUC). The balanced accuracy is defined as

$$\text{Bacc} = \frac{\sum_{c=1}^{C} (T_c / n_c)}{C}, \tag{13}$$

with T_c being the number of correctly classified bags of class c and n_c the number of bags in class c, for $c = 1, \ldots, C$.

The choice of metric depended on the dataset and the metric used in prior work (i.e., by selecting the same criterion, comparison with other works was possible). We performed a series of experiments by comparing different classifiers on the same datasets using 10-fold cross-validation and report the average accuracy. For each fold, we internally used a 2-fold cross-validation procedure on the training set in order to tune the hyperparameters of each method. Once the best parameters were determined, they were used to classify the test set to record the test accuracy. Therefore, all methods were assessed on independent test sets not used during training of the classification models, nor during the optimization of the hyperparameters. For fairness, we performed grid search in each fold for finding the best parameters for each of the compared methods (our own as well as other state-of-the-art methods).

3. Results and Discussion

For the evaluation of our proposed algorithm, we employed two datasets: (i) the Breast Cancer UCSB Center for Bio-Image Informatics benchmark dataset [36] consisting of histopathology color images and (ii) multichannel recordings from the FrailSafe project [37] monitoring older people. In the next sections, we describe in brief these datasets and how they are represented by multidimensional arrays.

3.1. Data Sets

3.1.1. UCSB Breast Cancer Image Classification. The UCSB breast cancer dataset [36] consists of color histopathology images of 58 subjects of size 896×768 pixels taken from 32 benign and 26 malignant breast cancer patients. The classification problem of these images was formulated as an MIL problem first by Kandemir et al. [4] who segmented the images in 7×7 patches and extracted features from each patch. In an MIL setting, image patches are considered as instances and images as bags. In order to represent the dataset as a tensor in our approach, we also segment each image in $p \times p$ patches and vectorize the pixels of each patch per channel ending up to a matrix where the rows of the matrix represent the pixels and the 3 columns represent the RGB channels. If we arrange all these matrices across the first dimension, we obtain a tensor of dimensions $I \times J \times 3$, where $I = 58 * p^2$ and J is the number of pixels per patch. If we devote the first mode of the tensor to the instances, the second mode to the pixels, and the third mode to the RGB channels, we end up with a 3-mode tensor, containing all instances as described earlier.

3.1.2. Physiological Signals from Monitoring Older People. This data set was collected as part of the FrailSafe project [37] and consists of physiological measurements acquired from older people (age > 70 years). The measurements are acquired during ordinary all day indoor or outdoor activities. The ultimate goal is to predict aging-associated decline in reserve and function (denoted as *frailty*) through the extraction of geriatric indices from multiparametric data. Standard frailty indices, such as the Fried phenotype of frailty [38], are based on the common geriatric assessment (performed

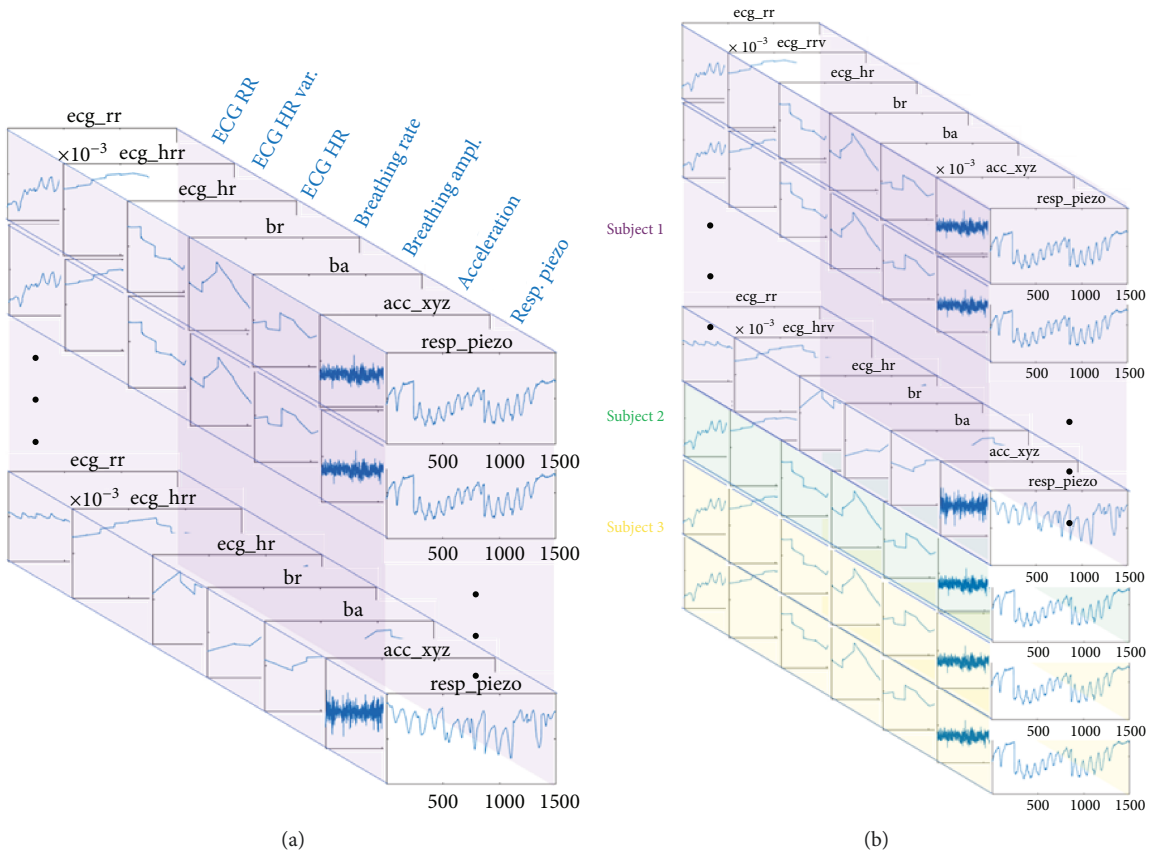

Figure 3: 3D-tensor for one subject (a) and 3D-tensor of all subjects (b).

sporadically and if considered necessary) and do not continuously monitor the health status, neither capture different medical domains. On the contrary, our goal is to extract frailty indicators from the multidimensional recordings in an effort to unobstructively monitor the health status of the older people. We assess the predictive power of physiological signals using TensMIL and the Fried score as ground truth, measured on the same time period with the acquired data. According to the Fried scale [38], three frailty stages can be distinguished: nonfrail, prefrail, and frail.

The physiological signals used in this study included time-synchronized measurements (calculated by dedicated software algorithms) from respiration, heart, posture, and physical activity. Seven channels were resampled at the same frequency (25 Hz): respiratory raw signal (by the piezoresistive sensor), magnitude of acceleration in 3 axes, breathing amplitude, breathing rate, ECG heart rate, ECG heart rate variability, and ECG RR interval. The measurements are recorded using two different devices, a fact that makes this dataset especially challenging. More details on the problem objective and the incorporated devices can be found in [1, 2].

The data representation in a tensorial form included the extraction of nonoverlapping time windows of one minute duration (i.e., 1500 time points). We consider the measurements in each time window for each subject as an instance, while the total recordings (all instances) for each subject compose one bag. In order to model the data in the form of a multidimensional array, we concatenate the multiple instances (i.e., time windows) of each subject in a 3-dimensional tensor $\mathcal{X}^{(i)}$ of dimensionality $n_i \times 1500 \times 7$, $i = 1, 2, \ldots, n$, where n_i is the number of instances available for each subject. In order to construct the whole tensor, we concatenate all tensors $\mathcal{X}^{(i)}$ along the first dimension to produce a new 3D-tensor $\mathcal{X}$ containing all instances of all bags as shown in Figure 3, resulting to a $19244 \times 1500 \times 7$ tensor. In Table 1, we summarize the available data per frailty group.

Table 1: Number of instances and percentages of bags (subjects) and instances (time windows) per class.

Class	Nr. of bags	Nr. of instances	Perc. of bags	Perc. of instances
Nonfrail	49	7127	42.24%	37.03%
Prefrail	54	8803	46.55%	45.74%
Frail	13	3314	11.21%	17.22%
Sum	116	19,244	100%	100%

3.2. Experiments

3.2.1. PARAFAC Feature Insights. Before proceeding with the results of the analysis, we provide some insights on the nature of the extracted features. As stated before, a tensor with full or missing values can be decomposed into R rank-1 components, producing a high-order dictionary that represents the

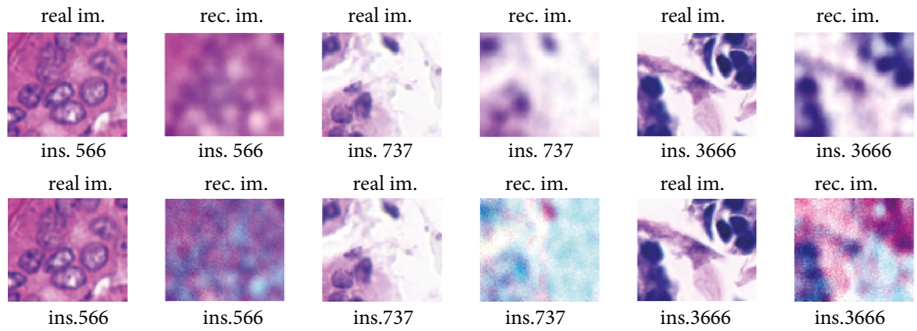

Figure 4: Random patches from BC images and their reconstruction with full values (upper row) using ALS and from 10% observed values using StrProxSGD (lower row).

latent concepts in the data. Since instances are assigned to the first dimension of the tensor, each mode-1 slice corresponds to an instance. Having computed the PARAFAC factors **U**, **V**, and **W**, we can compute, based on (1), the reconstruction of the data tensor either from full observed values or from a subset of the tensor's values (missing values).

Figure 4 depicts five random instances of the Breast Cancer dataset and their corresponding reconstructions with the ALS algorithm using full values (upper row) or with the StrProxSGD algorithm using 10% observed values (lower row). It can be observed that the reconstruction from full values results to a clearer version of the original images. As will be discussed in the next section, our experiments showed that the information preserved from the decomposition (even when using only 10% of the observed values) is sufficient to accurately classify the images in benign and malignant cases. The PARAFAC decomposition produces spatial ($\boldsymbol{V}$ from Equation (3)) and color ($\boldsymbol{W}$ from Equation (3)) components that correspond to the second and third dimension of the data tensor, which constitute the high-order dictionary. Figure 5 illustrates 40 (selected out of 120) spatial components of the dictionary. We observe also that the spatial components computed from 10% observed values are slightly noisier than the components computed from full values, a fact that showed to not significantly affect the classification accuracy.

3.2.2. Classification Assessment. The evaluation metric that we used for our experiments was different for each dataset. For the BC dataset, we report the AUC, since this metric was used for evaluation in the majority of other works. For the sake of completeness, we report also the mean test accuracy over 10 different test sets.

As reported in Table 1, the physiological signals dataset is highly unbalanced containing 11.21% of frail bags and about 42% and 47% of nonfrail and prefrail bags, respectively. For this reason, along with the test accuracy, we report also the balanced accuracy.

3.2.3. Breast Cancer Diagnosis from Histopathology Images. In this experiment, we computed the accuracy and the AUC of the proposed method against state-of-the-art MIL algorithms. We report results for each of the algorithms employing the features extracted by Kandemir et al. [4], and features extracted by the proposed method computing the PARAFAC decomposition from full values using the ALS algorithm [30] and from 10% randomly selected observed values using the StrProxSGD algorithm [31]. We should note here that the features extracted by Kandemir et al. [4] are application-specific in contrast to our extracted features that are problem-independent and can be obtained directly from any raw multidimensional data with the same procedure.

As can be observed in Table 2, when we employ the features from [4], our method is as good as JC2MIL [40] but it is outperformed by the other methods. This suggests that the feature extraction process is strongly related with the proposed MIL classification method. Indeed, when we employ the proposed features from tensor decomposition, performance improves as can be shown from the performance of TensMIL from full and 90% missing values, respectively. When using ALS features from full data, our method outperforms all other methods in terms of AUC, improving the performance by 4%–11% while in terms of accuracy TensMIL outperforms all other investigated methods and is comparable to MCILBoost. Overall, our method is comparable or outperforms other methods in terms of AUC and outperforms all other methods in terms of accuracy, except MILBOOST [41]. Concerning the case of data with missing values, our method outperforms in terms of accuracy all other investigated methods and in terms of AUC all methods except of JC2MIL to which it is comparable. Let us note here that the extraction of the hand-crafted features in [4] cannot be currently reproduced for data with missing values because the code for the feature extraction is not provided. Thus, for the missing values experiment, we compare only with the features extracted by StrProxSGD [31].

3.2.4. Physiological Signals for Frailty Prediction. In the next experiment, we evaluated the accuracy of TensMIL for frailty status prediction of older people based on motion, cardiac, and respiratory signals. In these experiments, the hyperparameters of the method were estimated by cross-validation on the training set (using the StrProxSGD algorithm for extracting features from 10% observed values) and were subsequently used for the case of full values. We performed two series of experiments. In the first experiment, we considered the three distinct frailty stages proposed by Fried (nonfrail, prefrail, and frail), whereas in the second experiment, we

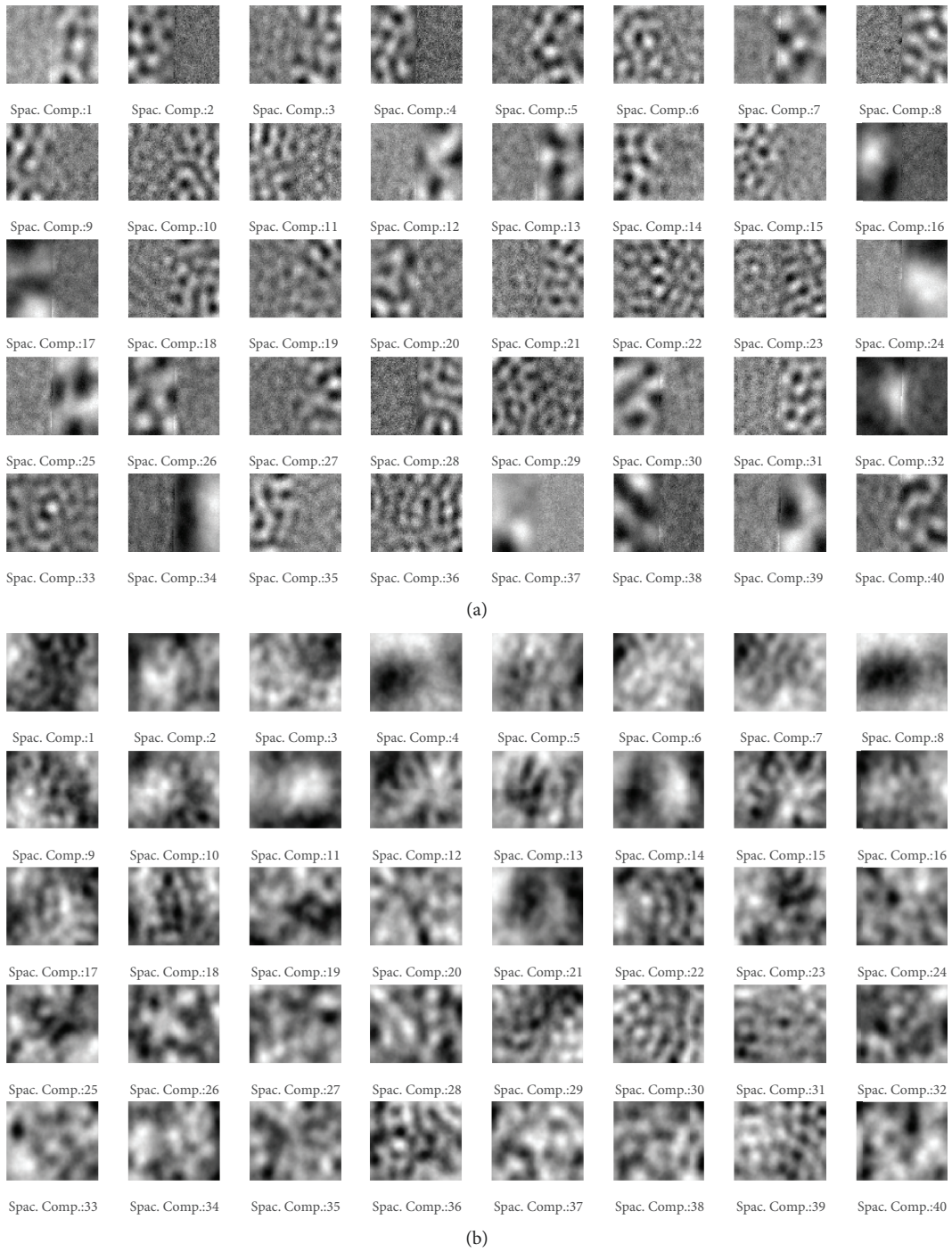

Figure 5: First 40 (out of 120) spatial components of PARAFAC model from 10% observed values using StrProxSGD (a) and from full values using ALS (b).

merged the prefrail and frail classes to create a less unbalanced dataset. Feature extraction was performed using the ALS algorithm from full data and the StrProxSGD algorithm for missing data. The results of the three class problem from full and incomplete data are shown in Table 3. When full values are considered, the accuracy of the proposed method is 45.76% (37% higher than the probability of random guess) and the balanced accuracy is 34.06% (similar to random guess). In contrast, when only 10% of the values are employed, we obtain accuracy 73.41% and balanced accuracy 67.17%, which is an improvement by a factor of 1.6 (for the accuracy) and 1.97 (for the balanced accuracy). These results strongly suggest that the data are highly noisy. Even though PARAFAC decomposition is robust against noise [43], ALS algorithm using full data could not find a good high-order dictionary for discrimination between the three classes. On the other hand, when only 10% of the data are employed, StrProxSGD could calculate a more suitable dictionary for

Table 2: Tenfold cross-validation mean test accuracy and mean AUC for the BC dataset.

BC	Kandemir [4]		ALS $R = 120$		StrProxSGD $R = 120$ (90% missing values)	
	Acc	AUC	Acc	AUC	Acc	AUC
MILES [39]	81.33 (0.15)	**0.91** (0.15)	72.67 (0.21)	0.79 (0.21)	63.33 (0.18)	0.72 (0.15)
JC2MIL [40]	74.33 (0.16)	0.84 (0.16)	72.33 (0.18)	0.78 (0.18)	77.67 (0.08)	**0.88** (0.14)
MILBoost [41]	**89.33** (0.09)	**0.94** (0.09)	81.67 (0.21)	0.87 (0.19)	68.33 (0.3)	0.77 (0.27)
MCILBoost [42]	82.33 (0.15)	**0.93** (0.12)	**85.00** (0.12)	**0.90** (0.12)	76.67 (0.22)	0.84 (0.16)
TensMIL	74.33 (0.16)	0.86 (0.16)	**84.67** (0.17)	**0.90** (0.15)	**79.33** (0.16)	0.85 (0.15)

Table 3: Test accuracy and balanced accuracy from full and 90% missing values for the 3 class problem.

Method	ALS $R = 60$		StrProxSGD $R = 60$ (90% missing values)	
	Acc	Bacc	Acc	Bacc
TensMIL	45.76 (0.13)	34.06 (0.09)	73.41 (0.01)	67.17 (0.13)

Table 4: Test accuracy from full and 90% missing values for the 2 class problem.

Methods	ALS $R = 60$	StrProxSGD $R = 60$ (90% missing values)
MILES [39]	51.59 (0.13)	67.20 (0.11)
JC2MIL [40]	**56.82** (0.07)	55.30 (0.08)
MILBoost [41]	50.83 (0.15)	54.39 (0.15)
MCILBoost [42]	45.46 (0.14)	60.91 (0.22)
TensMIL	**54.02** (0.13)	**80.83** (0.16)

the classification task. Let us note here that we do not report results from other MIL classifiers, since their performance was very poor when using the one-against-all strategy for the above multiclass problem.

Since the prefrail class lies between the frail and nonfrail class and in order to construct a more balanced dataset, we merged the prefrail with the frail group and examined the binary classification problem. As reported in Table 4, TensMIL achieved from 26.44% to 13.63% higher accuracy than the other methods by using only 10% of randomly selected values. For the case of full values, the proposed method achieves from 8.56% to 2.43% better accuracy. Only JC2MIL achieves slightly better accuracy than TensMIL.

In Table 5, we report also the mean CPU running time (across the 10-fold cross-validation sets) of TensMIL as compared to the other investigated state-of-the-art methods. The time reported corresponds to the frailty classification problem based on physiological signals, since this dataset was the largest among the two examined applications. The feature extraction component using tensor decomposition is the most time-consuming part of the method (it requires about 2.25 hours), whereas the MIL component is computationally fast. Specifically, the classification component in TensMIL requires 7 to ~52 times less training time as compared to the investigated classifiers. This fact is due to the simplicity of TensMIL since only a full quadratic regression and a QDA model have to be trained. In terms of the inference time (after feature extraction), TensMIL along with JC2MIL achieves a testing time under 1 second, which is faster than all the other investigated algorithms. We should note here that the experiments for the tensor decomposition were conducted on a Red Hat Enterprise Linux, release 6.7 (Santiago) server, comprising 162.8 GHz AMD Opteron™ 6320 processors with 62 Gb RAM, running MATLAB R2018a, while the experiments for measuring the training and test time were conducted on an Ubuntu 16.04 LTS desktop, comprising 42.0 Gz Intel® Xeon® CPU E5504 processors with 23.5 Gb RAM, running MATLAB R2017a.

Table 5: Mean CPU running time over the 10 cross-validation folds for the MIL classification component.

Methods	Training time[a]	Testing time[a]
MILES [39]	42 sec	1 sec
JC2MIL [40]	56 sec	<1 sec
MILBoost [41]	52 sec	5 sec
MCILBoost [42]	309 sec	6 sec
TensMIL	**6 sec**	**<1 sec**

[a]The experiments were conducted on an Ubuntu 16.04 LTS desktop, comprising 4 2.0 GHz Intel (R) Xeon (R) CPU E5504 processors with 23.5 Gb RAM, running MATLAB R2017a.

Finally, we compared our method with a clustering approach proposed in [1] for prediction of several clinical metrics that used statistical features from the same physiological signals, as well as other devices (GPS, game platform). Although this approach [1] showed high potential for some clinical metrics, the accuracy for the frailty index expressed by the Fried score was only 51% for the 2 class problem (nonfrail vs. prefrail and frail). TensMIL achieves 3.02% and 29.83% higher accuracy when all values or only 10% of the values are used, respectively. The clustering approach in [1] was not evaluated with missing values; however, we expect small deviations in accuracy due to the large time scale used for feature extraction and the statistical nature of the implemented features.

4. Conclusions

In this work, we exploited the high-order structure of health data through tensor decomposition aiming at extracting application-independent features that can facilitate prediction in multiple-instance learning paradigms. The prediction

models were trained in a sequential fashion to learn local and global content, while external hyperparameters were estimated by Bayesian optimization, thus providing an end-to-end architecture. The method could successfully represent and classify data with a significant amount (90%) of missing values. It was evaluated in the UCSB breast cancer benchmark dataset, as well as for prediction of aging-associated decline. In both application scenarios, the proposed method outperformed or was comparable to existing state-of-the-art machine learning techniques. Moreover, the obtained results were superior to our previous work based on statistical features and cluster analysis. Future work includes the investigation of sparse representations and addition of nonnegativity and orthogonality constraints for the extraction of more natural and interpretable data concepts.

Data Availability

The UCSB Breast Cancer data set is publically available and can be downloaded from https://bioimage.ucsb.edu/research/bio-segmentation. The data of the physiological signals for frailty prediction are collected as part of the FrailSafe Project [27] and will be available at the repository of the project: https://frailsafe-project.eu/ (contact:vasilis@ceid.upatras.gr).

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The research reported in the present paper was partially supported by the FrailSafe Project (H2020-PHC-21-2015-690140) "Sensing and predictive treatment of frailty and associated co-morbidities using advanced personalized models and advanced interventions" cofunded by the European Commission under the Horizon 2020 research and innovation program. The authors want to thank all ICT (Smartex, CERTH, Gruppo Sigla) and medical partners from the FrailSafe Project for data sharing and annotations. They especially wish to thank their colleagues K. Deltouzos and S. Kalogiannis for the help with data preprocessing.

Supplementary Materials

The preliminaries of tensors and their rank decompositions. *(Supplementary Materials)*

References

[1] S. Kalogiannis, E. I. Zacharaki, K. Deltouzos et al., "Geriatric group analysis by clustering non-linearly embedded multi-sensor data," in *2018 IEEE International Conference on Innovations in Intelligent Systems and Applications (INISTA 2018)*, Thessaloniki, Greece, 2018.

[2] A. Papagiannaki, E. I. Zacharaki, K. Deltouzos et al., "Meeting challenges of activity recognition for ageing population in real life settings," in *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*, pp. 1–6, Ostrava, Czech Republic, 2018.

[3] G. Lu, L. Halig, D. Wang, X. Qin, Z. G. Chen, and B. Fei, "Spectral-spatial classification for noninvasive cancer detection using hyperspectral imaging," *Journal of Biomedical Optics*, vol. 19, no. 10, article 106004, 2014.

[4] M. Kandemir, C. Zhang, and F. A. Hamprecht, "Empowering multiple instance histopathology cancer diagnosis by cell graphs," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2014. MICCAI 2014. Lecture Notes in Computer Science, vol 8674*pp. 228–235, Springer, Cham.

[5] K. Mosaliganti, F. Janoos, O. Irfanoglu et al., "Tensor classification of *N*-point correlation function features for histology tissue segmentation," *Medical Image Analysis*, vol. 13, no. 1, pp. 156–166, 2009.

[6] V. G. Kanas, E. I. Zacharaki, E. Pippa, V. Tsirka, M. Koutroumanidis, and V. Megalooikonomou, "Classification of epileptic and non-epileptic events using tensor decomposition," in *2015 IEEE 15th International Conference on Bioinformatics and Bioengineering (BIBE)*, Belgrade, Serbia, November 2015.

[7] C.-F. V. Latchoumane, F. B. Vialatte, J. Solé-Casals et al., "Multiway array decomposition analysis of EEGs in Alzheimer's disease," *Journal of Neuroscience Methods*, vol. 207, no. 1, pp. 41–50, 2012.

[8] A. Cichocki, D. Mandic, L. de Lathauwer et al., "Tensor decompositions for signal processing applications: from two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, no. 2, pp. 145–163, 2015.

[9] N. D. Sidiropoulos, L. de Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos, "Tensor decomposition for signal processing and machine learning," *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551–3582, 2017.

[10] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A survey of multilinear subspace learning for tensor data," *Pattern Recognition*, vol. 44, no. 7, pp. 1540–1551, 2011.

[11] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H. J. Zhang, "Multilinear discriminant analysis for face recognition," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 212–220, 2007.

[12] X. Zhang, X. Yuan, and L. Carin, "Nonlocal low-rank tensor factor analysis for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, UT, USA, June 2018.

[13] B. Du, M. Zhang, L. Zhang, R. Hu, and D. Tao, "PLTD: patch-based low-rank tensor decomposition for hyperspectral images," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 67–79, 2017.

[14] Y. Wang, L. Lin, Q. Zhao, T. Yue, D. Meng, and Y. Leung, "Compressive sensing of hyperspectral images via joint tensor Tucker decomposition and weighted total variation regularization," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 12, pp. 2457–2461, 2017.

[15] A. S. Lalos, I. Nikolas, E. Vlachos, and K. Moustakas, "Compressed sensing for efficient encoding of dense 3D meshes using model-based Bayesian learning," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 41–53, 2017.

[16] Y. Wang, D. Meng, and M. Yuan, "Sparse recovery: from vectors to tensors," *National Science Review*, vol. 5, no. 5, pp. 756–767, 2018.

[17] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations with missing data," in *2010 SIAM International Conference on Data Mining*, pp. 701–712, Columbus, OH, USA, April-May 2010.

[18] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing and Applications*, vol. 19, no. 2, pp. 263–282, 2010.

[19] G. Andrew, B. Recht, J. Xu, R. Nowak, and X. Zhu, "Transduction with matrix completion: three birds with one stone," in *Advances in Neural Information Processing Systems 23 (NIPS 2010)*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds., pp. 757–765, Curran Associates, Inc., 2010.

[20] E. Hazan, R. Livni, and Y. Mansour, "Classification with low rank and missing data," in *Proceedings of the 32 nd International Conference on Machine Learning*, Lille, France, 2015.

[21] D. Porro-Muñoz, R. P. W. Duin, and I. Talavera, "Missing values in dissimilarity-based classification of multi-way data," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2013. Lecture Notes in Computer Science, vol 8258*, J. Ruiz-Shulcloper and G. Sanniti di Baja, Eds., Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[22] J. Amores, "Multiple instance classification: review, taxonomy and comparative study," *Artificial Intelligence*, vol. 201, Supplement C, pp. 81–105, 2013.

[23] J. Foulds and E. Frank, "A review of multi-instance learning assumptions," *The Knowledge Engineering Review*, vol. 25, no. 1, pp. 1–25, 2010.

[24] L. Dong, *A Comparison of Multi-instance Learning Algorithms*, University of Waikato. The University of Waikato, Hamilton, New Zealand, 2006.

[25] X. Xu, *Statistical Learning in Multiple Instance Problems*, University of Waikato. The University of Waikato, Hamilton, New Zealand, 2003.

[26] N. Weidmann, E. Frank, and B. Pfahringer, "A two-level learning method for generalized multi-instance problems," in *Machine Learning: ECML 2003. ECML 2003. Lecture Notes in Computer Science, vol 2837*pp. 468–479, Springer, Berlin, Heidelberg.

[27] J. Wu, S. Pan, X. Zhu, C. Zhang, and X. Wu, "Multi-instance learning with discriminative bag mapping," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 6, pp. 1065–1080, 2018.

[28] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, 1970.

[29] R. A. Harshman, "Foundations of the PARAFAC procedure: models and conditions for an "explanatory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.

[30] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.

[31] T. Papastergiou and V. Megalooikonomou, "A distributed proximal gradient descent method for tensor completion," in *2017 IEEE International Conference on Big Data (Big Data)*, Boston, MA, USA, December 2017.

[32] C. Leistner, A. Saffari, and H. Bischof, *MIForests: Multiple-Instance Learning with Randomized Trees*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[33] W. Dumouchel and F. O'Brien, "Integrating a robust option into a multiple regression computing environment," in *Computing and graphics in statistics*, pp. 41–48, Springer-Verlag New York, Inc., New York, NY, USA, 1991.

[34] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer Series in Statistics, Springer-Verlag New York, 2009.

[35] M. A. Gelbart, J. Snoek, and R. P. Adams, "Bayesian optimization with unknown constraints," in *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pp. 250–259, AUAI Press, Quebec City, Quebec, Canada, 2014.

[36] E. D. Gelasca, J. Byun, B. Obara, and B. S. Manjunath, "Evaluation and benchmark for biological image segmentation," in *2008 15th IEEE International Conference on Image Processing*, San Diego, CA, USA, October 2008.

[37] "Frail safe project," Available from: https://frailsafe-project.eu/.

[38] L. P. Fried, C. M. Tangen, J. Walston et al., "Frailty in older adults evidence for a phenotype," *The Journals of Gerontology: Series A*, vol. 56, no. 3, pp. M146–M157, 2001.

[39] Y. Chen, J. Bi, and J. Z. Wang, "MILES: multiple-instance learning via embedded instance selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 1931–1947, 2006.

[40] K. Sikka, R. Giri, and M. S. BartlettX. Xie, M. W. Jones, and G. K. L. Tam, "Joint clustering and classification for multiple instance learning," in *Proceedings of the British Machine Vision Conference (BMVC)*, BMVA Press, 2015.

[41] P. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Proceedings of the 18th International Conference on Neural Information Processing Systems*, pp. 1417–1424, MIT Press, Vancouver, British Columbia, Canada, 2005.

[42] Y. Xu, J. Y. Zhu, E. I. C. Chang, M. Lai, and Z. Tu, "Weakly supervised histopathology cancer image segmentation and classification," *Medical Image Analysis*, vol. 18, no. 3, pp. 591–604, 2014.

[43] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, "Scalable tensor factorizations for incomplete data," *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41–56, 2011.