Discovery of temporal patterns from multivariate time series data to support the classification of dementia profiles

José Mota[1,2]
[1]INESC-ID and Instituto Superior Tecnico, Lisbon, Portugal
[2] jose.m.mota@tecnico.ulisboa.pt

Abstract—**Data Mining has been used in the Medical field as an auxiliary method to support diagnostics and prognostics. However, the current methods are limited in their ability to deal with the high multivariate order and inherent temporal nature of biomedical data. The present work aims to study the role of discriminative temporal patterns in the classification of Multivariate Time Series (MTS) data with the aim of diagnosing the different phases of the Alzheimer's Disease (AD). To this end, we use data extracted from the Alzheimer's Disease Neuroimaging Initiative (ADNI), clinical trial data for the prevention and treatment of AD. In this context, this work analysis heterogeneous data sources from the ADNI database obtained along periodic assessments, including biomarkers, cerebrospinal fluid indicators, neuropsychological tests and features extracted from structural Magnetic Resonance Imaging (MRI). In the proposed study, we apply different techniques, including triclustering and associative classification for the analysis of multivariate time series data derived from ADNI. Gathered results outperformed the implemented baselines approaches, highlighting the importance of using discriminative temporal patterns in associative MTS classifiers, and in particular, offer empirical evidence that confirms their role in supporting clinical diagnostics.**

I. Introduction

Alzheimer's Disease is a degenerative brain disease and the most common cause of dementia, which is characterised by a decline of cognitive skills, lack of memory, communication and language issues, loss of the ability to focus, pay attention, reasoning, judgment, and visual perception [1].

AD prevalence has shown significant growth. The last study done by the WHO points out that, nowadays, the worldwide prevalence of dementia is approximately 50 million and is estimated that this number could reach 82 million in 2030 and 152 million in 2050, with AD representing the most significant portion of this number [2].

During the past decades, studies were limited by the reduced number of available data. Therefore, significant efforts were made to build several repositories which are available nowadays. Despite the abundance of data, other issues have emerged: 1) the need for new strategies to handle high quantities of data; and 2) the need to integrate heterogeneous views on human health (modalities). Recently, new data mining algorithms have been proposed to improve the knowledge of disease's mechanisms by combining the abundance of both biological and medical data [3]. These studies yield new ways to support the clinical understanding, diagnosis and prognosis of dementia. Nonetheless, some gaps have to be filled, in order to obtain a full comprehension of the disease's progression. The present study is designed to fill some of those gaps, thus improving the understanding and personalised prediction of dementia progression.

The present study aims to analyse heterogeneous data collected by ADNI. This initiative aims to identify biomedical markers of the Alzheimer's disease and contribute to the understanding and prognosis of this pathology. The data from this initiative is composed of clinical and biological indicators belonging to patients at different stages of dementia (strong memory complains, mild-cognitive impairment, Alzheimer disease).

Despite the existence of several state-of-the-art approaches, which are designed to learn classification models from Three-Dimensional (3D) data, none of them can handle the problem at hand [4] [5]. To this end, we present a novel approach to fill the gap that exists in the currently available methods. In this context, we propose the discovery of informative temporal patterns given by discriminative triclusters, followed by a calculus that estimates the likelihood of a given temporal pattern to be observed in a given patient. This proposal leads us to the presentation of a new approach of MTS classification.

Throughout this thesis, the proposed approach outperforms the implemented baselines, emphasising the discriminative power of the triclusters. Also, an ensemble approach, combining the features produced by all the approaches, was proposed. Results show that it is able to improve the performance of most classifiers, over the isolated approaches. Consequently, the promising results gathered on this work, demonstrate the crucial role of three-dimensional patterns in MTS data classification.

This work will be organised as follows. First, *Section 2* introduces some definitions about key topics of our work. Second, *Section 3* offers a compilation of related work. The following *Section* deals with the proposed solution. *Section 5* presents the results, finally, conclusions are drawn in *Section 6*.

A. *Research Contributions*

1) Computational Contributions: The accomplished computational goals include: Exploratory Data Analysis, Data Preprocessing, Discriminative Pattern Discovery, Associative Classification based on Patterns, Classic Classification, and Statistical Testing of different hypothesis.

2) Biomedical Contributions: Results offer initial evidence that additional biomedical-specific goals can be further reached, including: the comprehensive description and discrimination of AD's phases based on multivariate time series data; the discovery of non-trivial patterns of disease progression; the identification of biomedical markers essential for supporting the AD diagnosis; to validate the prognostic models of AD.

3) Hypothesis: The thesis' hypothesis is: *non-trivial temporal patterns improve the ability of associative classifiers to label MTS data*. This hypothesis will be particularly validated in the context of ADNI clinical trials in order to expand the understanding of the progression of AD.

II. Background

Alzheimer's Disease. AD is a neurodegenerative disease and, currently, the most common cause of dementia. [1].

Alzheimer's Disease Neuroimaging Initiative. ADNI is a multisite study that aims to improve clinical trials for the prevention and treatment of AD.

Mild Cognitive Impairment. MCI participants are a control group in the ADNI study, who have diagnosed a subjective memory concern, however, daily living activities are preserved [6].

Cognitively Normal. CN participants are a second control group in the ADNI study, who show no signs of depression, mild cognitive impairment, or dementia [6].

Classification. Given a set of observations $\vec{X} = \{\vec{x}_1, .., \vec{x}_n\}$ and a set of labels $\Sigma = \{c_1, c_2, \ldots, c_n\}$, a classifier is a model M that maps the data space of observations into the set of labels in order predict the label of a new observation ($\vec{x}_{new}$): $\hat{z} = M(\vec{x}_{new})$, where $\hat{z} \in \Sigma$ [7].

K-fold Cross-Validation. Given a dataset D, K-fold Cross-Validation is the process of split D into K folds with the same size. Each fold is treated as the testing set and the remaining folds are treated as the training set. Then, the process is completed k times, for each fold [7]

Multivariate time series data. MTS data of p order and m length $\vec{A}$ is defined by n observations $\vec{X} = \{\vec{x}_1, .., \vec{x}_n\}$, p attributes $\vec{Y} = \{\vec{y}_1, .., \vec{y}_p\}$, and m time points $\vec{T} = \{\vec{t}_1, .., \vec{t}_m\}$. An element a_{ijk} is a value measured on observation $\vec{x}_i$, attribute $\vec{y}_j$ and time point $\vec{t}_k$ [8].

MTS data. MTS data can be *real-valued* ($a_{ijk} \in R$), *symbolic* ($a_{ijk} \in \Sigma$ and Σ is a set of nominal or ordinal symbols), *integer* ($a_{ijk} \in Z$), or *non-identically distributed* ($a_{ijk} \in \mathcal{A}_j$ where $\mathcal{A}_j$ is the domain of the $\vec{y}_j$'s attribute) [8].

Tricluster. Given a MTS dataset, also referred as three-dimensional (3D) dataset, $\vec{A}$ with n observations $\vec{X}$, m attributes $\vec{Y}$, and p timepoints $\vec{T}$, a tricluster $\mathcal{T} = (\vec{I}, \vec{J}, \vec{K})$ is a subspace of the original space, where $\vec{I} \subseteq \vec{X}, \vec{J} \subseteq \vec{Y}$, and $\vec{K} \subseteq \vec{\mathcal{ABC}}$ are subsets of observations, attributes and timepoints, respectively [8].

Triclustering. Triclustering task aims to find a set of triclusters in order that each tricluster satisfies specific criteria of homogeneity and statistical significance, where homogeneity is defined as coherence criteria need to be placed under the target problem and desirable outputs. In addition, a statistically significant tricluster is a tricluster whose probability to occur (against a null model) is unexpectedly low [8].

III. Related Work

This section presents relevant works on triclustering and MTS classification.

A. *Triclustering*

The task of clustering aims to create subgroups that show greater similarity compared to the original group. In our context, we propose the discovery of discriminative temporal patterns, using triclustering algorithms.

The unsupervised analysis of three-dimensional data has been applied to unravel biological and physiological patterns underlying disease progression [8]. One of the most significant gaps of biclustering, the discovery of subspaces in tabular data, is that it disregards the third dimension, and Triclustering tackles that obstacle. In our problem, the periodic collection of biological information and the clinical monitoring of patients are an example of a multivariate time series dataset, patient-record-time. Henriques et al. [8] described different ways to discover and assess the quality of triclusters using merit functions. A constant tricluster $(\vec{I}, \vec{J}, \vec{K})$ has a mean of values, $a_{ijk} = c + \eta_{ijk}$, where c is the expected value and η_{ijk} is the noise factor. Another way to find constant triclusters is by using the variance of theirs values:

$$\sigma^2 = \frac{1}{\vec{I}\vec{J}\vec{K}} \sum_{i\in\vec{I}} \sum_{j\in\vec{J}} \sum_{k\in\vec{K}} (a_{ijk} - a_{IJK})^2, \tag{1}$$

where a_{IJK} is the mean of all values of a tricluster.

An additional merit function is the residue-based function, which ensures more flexible forms of homogeneity. The elements of a tricluster are defined as:

$$a_{ijk} = c + \alpha_i + \beta_j + \gamma_z + \eta_{ijk}, \tag{2}$$

where α_i is the contribution from the observation x_i, β_j is the contribution from the attribute y_j and, finally γ_z is the contribution from the timepoint z_k. In this context, we can apply the Mean Squared Residue (MSR) to manage the triclustering algorithm. The MSR of a subspace is the average of the squared residues, which is defined as:

$$MSR_{\vec{I}\vec{J}\vec{K}} = \frac{1}{IJK} \sum_{i\in\vec{I}} \sum_{j\in\vec{J}} \sum_{k\in\vec{K}} \eta_{ijk}, \tag{3}$$

where $\eta_{ijk} = a_{ijk} - (a_{iJK} + a_{IjK} + a_{IJk} - 2a_{IJK})$.

Besides the 3D merit functions described above, there are two additional approaches: 1) 2D Merit Functions (biclustering) followed by Consensus, and 2) Pattern-based merit functions. The first one aims to discover the correlation in two-dimensional slices of a 3D dataset and then search for consensus across the third dimension. The second one tries to find triclusters with clear patterns. More specifically, this approach aims to maximise the volume of a tricluster respecting the given pattern.

Henriques et al. [8] further surveyed different triclustering approaches:

Greedy Approaches. The greedy approach proposes to maximise the 3D merit function described above in equation (3). Bhar et al. [9] proposed the algorithm $\delta - TRIMAX$ which attempts to find largest and maximal triclusters in a 3D dataset.

In addition, Liu et al. [10] proposed an algorithm that extends the biclustering method to three-dimensional data spaces. This algorithm starts with a randomly initial tricluster and parameters. For each iteration, the algorithm computes the regulated expression values score of every possible two-dimensional region in the entire subspace and will stop when reaching maximal iteration. Moreover, it uses the Automatic Boundary Scanning algorithm to obtain a value of the threshold, to be robust to outliers.

Biclustering-based Approaches. Jiang et al. [11] proposed a novel algorithm to find 3D clusters on Gene-Sample-Time (GST) microarray data, called gTRICLUSTER. This proposal intends to find biclusters in data stored as $sample \times time$ and, then mining triclusters by joining biclusters on the gene dimension. The similarity metric chosen by the authors was the Spearman Rank Correlation (SRC), which is more flexible and robust to noise than in previous triclustering algorithms [12]. Firstly, the algorithm constructs the matrix of similarity between genes. Afterwards, gTRICLUSTER applies a depth-first search in order to list the possible maximal cliques and consequently identify candidate biclusters. Then, to obtain 3D clusters, for each subset of sets that are found, it chooses subsets with maximal coherence. Despite its strengths, Henriques et al. [8] concluded that this algorithm can be biased due to sample and gene size dominance because it ignores intergene coherence.

Pattern-based Approaches. Liu et al. [13] proposed a novel framework that explores association rule mining on a 3D data derived from microarrays. They developed the 3D-TDAR-Mine that aims to identify triclusters using the temporal dependency association rule in the 3D dataset. The proposal can be described as follows. Firstly, Frequently Coherent Patterns (FCP) are identified for each gene (sample-time matrix). Secondly, the algorithm merges the FCP in the adjacent windows. Thirdly, after the merging, the method finds candidate rules that have temporal dependency. Finally, association rule mining is used to detect dependencies between patterns.

B. Multivariate time series (MTS) classification

Batal et al. [14] developed a temporal abstraction framework to label MTS for classification tasks. They proposed the Segmented Time series FeatureMine algorithm that automatically mines discriminative temporal abstraction patterns and then is used to learn a classification model. The algorithm starts with the Apriori algorithm applied in frequent pattern mining and aims to recognise the most common temporal patterns for each class. After these, the algorithm chooses the patterns that can better distinguish the target classes. The proposed methodology consists of the following steps:

1) Segment the time series in order to obtain a qualitative representation of the raw time series data;
2) Generate the frequent patterns from the segmented series of each class;
3) Select the frequent patterns, which are more frequent in one class and less frequent in other classes;
4) Transform the data into a vector format, which will be given to the classifier to build the classification model.

The methodology proposed by Batal et al. [14] integrates two powerful data mining paradigms: frequent pattern mining and inductive classification to solve the complex time series classification problem.

Complementarily, we can group MTS classifiers into one of four major approaches:

1) *Traditional classification* Firstly, there is a mapping of temporal data into tabular data by retrieving statistics along the time series and discarding temporal dependence between the statistics produced for each time window. Then, a classic classifier is applied [15].
2) *Distance-based classification* Time series distances are calculate to assess the similarity between observations. An illustrative distance metric is the Dynamic Time Warping (DTW), and a paradigmatic classifier is the k-nearest neighbors classifier [16].
3) *Associative time series classification (pattern-centric classifier).* Discovering discriminative temporal patterns from multivariate time series data and labeling a new observation based on the present patterns and the class they discriminate [17].
   Batal's algorithm [14] described above and our work fit in this major approach.
4) *Generative time series classification.* This approach includes generative models such as artificial RNN, DBN and HMM. Here, the best complex function is learnt to map the input time series and the output label [18] [19] [20].

IV. Solution

This thesis aims at exploring discriminative patterns observed along the different stages of the Alzheimer's disease and assesses whether they can enhance the classification of individuals. The target data is primarily given by multivariate time series extracted from the ADNI database. In particular, with our proposal, we intend to assess the value of triclustering algorithms for finding non-trivial patterns from temporal data. This aim is further challenged by the specificities of the targeted data from the ADNI initiative. First, the data contains both temporal and static variables. Second, it is an unbalanced dataset since the number of clinical trials per patient is not constant, as well as with regards to the number of individuals at the different stages of the disease. Third, the data has several missing values.

A. *Preprocessing*

Data gathering. This work focus on a dataset called ADNIMERGE, which contains a merge of several ADNI data sources. The ADNIMERGE dataset merges several of the key variables from various case report forms and biomarker lab summaries across the ADNI protocols. The dataset is composed of 13722 observations and contains 85 variables, including temporal and static variables. Temporal variables contain neurophysiological indicators extracted from MRI data and neuropsychological records given by cognitive assessments. In addition, static attributes include demographic data providing information about ADNI participants.

As we can see in Figure 1, the number of measurements per patients along time significantly varies. The original dataset was reshaped into a set of multivariate timeseries with an equal number of observations per patient. In accordance with the provided statistics, we decided to consider the range between the first and fifth measurement, monitoring each patient for two years and a half.

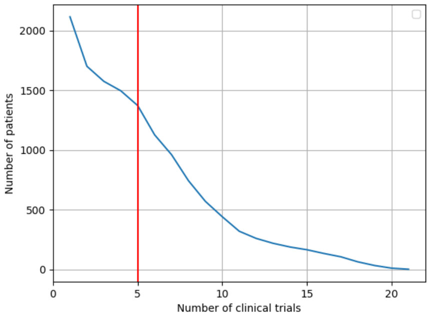


Fig. 1: Cumulative number of clinical trials per patients

The choice of this temporal window led to an analysis of 1362 individuals, where the first 5 time points of each individual are selected. It was concluded that the use of the first 5 time points (in contrast with other time windows) is the setting that allowed us to better exploit the different stages of Alzheimer's disease. Subsequently, the target class is defined as the concatenation of the disease stage at the first assessentment with the disease phase at the fifth one. Illustrating, given an individual with semiannual diagnostics {CN, CN, MCI, MCI, MCI}, the target class would be CN:MCI.

The resulting dataset is in fact a well-structured three-dimensional dataset, with 1362 patients, 85 features, and 5 time points. Figure 2 offers an abstract representation of the studied dataset, where each cube corresponds to either a numeric or categorical value.

Missing value imputation. The principles and techniques used for handling the missing data are:

1) *Class attribute* - Due to the absence of diagnostics in some of the individuals' assessments, we applied Last Observation Carried Forward (LOCF) [21]. This technique replaces every missing value with the last observed value, which corresponds to the last patients' diagnosis. In this case, we are assuming that the value of the outcome remains unchanged by the missing data. Illustrating, given an individual with semiannual diagnostics {MCI, MCI, AD, ∅, ∅ }, where ∅ is a blank value, it is assumed that the target class is MCI:AD.

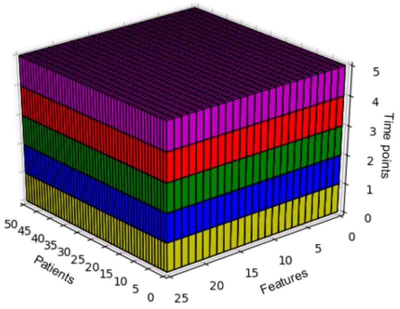


Fig. 2: Three-dimensional data: abstraction

2) *Numeric attributes* - The domain attributes with missing entries are numeric attributes. Considering we are studying a time series of five-time points, a Regression imputation was applied as a technique for handling the missing data. In regression imputation, the existing observation is used to make a prediction and then the predicted values replace the missing data. Due to the few time points, we adopt a linear regression, where the process is detailed in section IV-B. Additionally, the process is just applied if we had more than 3 out of 5 time points. Illustrating, given an individual with semiannual measurements $\{1, 2, 3, 4, \emptyset\}$, the blank value is replaced by 5.
3) *Candidate attributes for triclustering: filtering attribute with high fraction of missing values* - The triclustering searches [12] to be considered in our work do not support any missing values, consequently to do a further exploration of tricluster advantages in associative classification, we settle out that we only analyze the temporal features that had at least 70% values. This decision was due to triclustering algorithms detect coherence in data variation, an exhaustive imputation of missing value could lead to the formation of an nonexistent pattern. With the filtering of such features, triclustering searches algorithm are applied to 14 features of the total 85 features.
   In this step, we consider the mean substitution. "The theoretical background of the mean substitution is that the mean is a reasonable estimate for a randomly selected observation from a normal distribution" [21]. In this this approach, the mean value of a feature y_j at instance t is calculated for all patients in order to fill the missing value. With this procedure, we are not bias the model, and we are able to apply the triclustering algorithm.

B. Baseline window based approach

In this section, we introduce the two baseline approaches.

Simple approach As a first method to create a classification model using MTS, we propose the "simple approach". In this method, we reshape our three-dimensional dataset into a two-dimensional dataset, where the values from the time points of each feature are separated into a matrix. This example is called the "simple approach" since there is no abstraction of the temporal context. Consequently, a cube dataset with the dimensions $< 1362, 85, 5 >$ is transformed into a two-dimensional dataset with dimension $< 1362, 425 >$.

Regression-based approach As another baseline for our temporal classifier, we also present the regression-based approach, using linear regression. With this method, we are aiming to create a temporal abstraction, understanding the variety of data points through time.

As explained before, we intend to examine the five-time points for each temporal feature to extract from time series two features of interest: slope (β) and y-intercept (α). The α represents the predicted value of y_j, when $t = 0$. On the other hand, the slope represents the average predicted change in y_j resulting from a one-unit increase in t. In accordance with the criteria referred above, the regression-based approach applies a linear function which predicts the value of interest y_j assuming that it has a linear relationship with the remaining data points. Therefore, as an outcome of this proposal, we build a dataset composed of each patient's slope and y-intersect for each feature.

This process is iterated for all temporal features. Despite the existence of infinitely different functions passing through few data points (5), we decided to use linear regression, because it gives us a direct elucidation of temporal features behaviour over two and a half years.

Normality Test. We aim to demonstrate that the linear regression coefficients of temporal features for each dementia progression profile have distinct distributions. To this end, we applied a normality test, to guarantee that the sample of coefficients is well-modelled by a normal distribution. Actually, if we can verify that the samples of coefficients for each disease stage have significantly different normal distributions, then it is strong evidence of the discriminative capacity of this approach on MTS classification. To this end, we applied D'Agostino's K-squared normality test [22] whether the regression parameters were described by a Gaussian assumption.

Gathered results from D'agostino's K-squared test show that some parameters from linear regression have normality behaviour. Furthermore, we can prove that, for some feature, the population has different behaviours according to the target class.

Tricluster-based approach In this approach, unlike the baselines, we centre only on temporal features, which vary over time (two years and a half).

TRICLUSTER *Algorithm* With the propose of finding patterns along the three-dimensional data, Triclustering algorithms were applied to find subspace of the original space that has specific criteria of homogeneity and could discriminate difference stages of Alzheimer's disease.

We used the TRICLUSTER algorithm [12], which is up to date a state-of-the-art triclustering algorithm of reference, mining coherent clusters in 3D gene expression datasets (microarray dataset). In this contex, our shaped dataset could be comprehended as a microarray dataset, in which gene-sample-time data analysis becomes a patient-feature-time dataset. TRICLUSTER can mine arbitrarily positioned and overlapping biclusters including constant, scaling and shifting patterns from a 3D dataset. This algorithm has three main steps [8]:

1) Construction of a multigraph to store similar value ranges between all pairs of samples;
2) mining maximal biclusters from the multigraph formed for each time point (slices of the 3D dataset);
3) extracting triclusters by merging similar biclusters from different time points.

Optionally, the algorithm can delete or merge triclusters, depending on the parameterisation of the algorithm. We will further describe each step.

For labelled data, our primary purpose for the application of applying tricluring algorithms is to find discriminatory triclusters among classes and consecutively use the found patterns as a patient classification tool. To this end, our approach can be described as: 1) split the dataset into five three-dimensional samples one for each target class; 2) apply the TRICLUSTER algorithm for each class-conditional dataset and 3) validate and store the retrieved triclusters.

Assessing the discriminative power of triclusters After the phase of searching for three-dimensional patterns using the algorithm, it is necessary to find an effective way on how to use them in the associative classification model. The tricluster algorithm searches for patterns that satisfy specific criteria of homogeneity, using a three-dimensional sample. In this respect, it is challenging to create a metric to measure the similarity of a patient's observations (two-dimensional) with the found triclusters in the previous phase.

In the following paragraphs, we will explain how we faced this challenge.

Correlation approach for the discriminative power calculus. The problem displayed above led us to present a new approach to measure the similarity of variation of an isolated observation with the tri-dimensional found pattern. Considering we are working with numerical data, one possibility is the application of Pearson correlation.

Nevertheless, triclusters with two dimensions in the features are not candidates to apply the Pearson correlation. That is due to the Pearson correlation measures the linear relationship between two variables. As a result, two points always define a perfect linear relationship. Thus, the Pearson correlation coefficient between two points is always 1 or -1, which does not give to us any information about the problem at hand. Consequently, we chose the Cosine similarity as an alternative.

Cosine similarity Cosine similarity is a metric used to measure the similarity between two vectors, applying the inner product space which measures the cosine of the angle between them. Then, when the found tricluster has dimension two on features, we can transform the data into a vector and check the similarity between observations. The output is bounded in $[-1, 1]$. The value -1 means that the vectors are exactly opposite, 1 are the same, and finally, 0 indicates orthogonality.

We can derive this measure of similarity from the Euclidean dot product formula:

$$\vec{A} \cdot \vec{B} = \vec{A}\vec{B} \cos\theta, \tag{4}$$

We can conclude that:

$$similarity = \cos\theta = \frac{\vec{A} \cdot \vec{B}}{\vec{A}\vec{B}} = \frac{\sum_{i=0}^{i=n} A_i B_i}{\sqrt{\sum_{i=0}^{i=n} A_i^2}\sqrt{\sum_{i=0}^{i=n} B_i^2}}, \tag{5}$$

where A_i and B_i are components of each vector A and B respectively.

Pearson correlation coefficient With this approach, we intend to figure out if the data variation of the tricluster elements have a strong correlation and calculate the probability of a test observation belonging to the tricluster. As the approach described above, the Pearson correlation coefficient is a measure of the linear correlation between two variables. The coefficient is bounded in $[-1, 1]$, where 1 is total positive linear correlation, 1 is total negative linear correlation, and 0 is no linear correlation.

Calculating the correlation between two vectors, $\vec{A}$ and $\vec{B}$, the formula applied was:

$$r_{\vec{A}\vec{B}} = \frac{\sum_{i=0}^{i=n}(A_i - \bar{A}) \sum_{i=0}^{i=n}(B_i - \bar{B})}{\sqrt{\sum_{i=0}^{i=n}(A_i - \bar{A})^2}\sqrt{\sum_{i=0}^{i=n}(B_i - \bar{B})^2}}, \tag{6}$$

where $r_{\vec{A}\vec{B}}$ is correlation coefficient between $\vec{A}$ and $\vec{B}$, and A_i and B_i are components of each vector $\vec{A}$ and $\vec{B}$ respectively. Finally, $\bar{A}$ and $\bar{B}$ represents the sample mean of $\vec{A}$ and $\vec{B}$ respectively.

Implementation. In algorithm .1 is described the process. Our goal is to build a matrix with the Correlation value (ρ) of all patients for the components of found triclusters. Each value of the matrix corresponds to the mean of similarity coefficients between the data of the patient and the data from each tricluster slice (bicluster). First, we initialise an empty matrix $|I| \times |B|$, where $|I|$ is the number of patients and $|B|$ is the number of all triclusters slices.

Afterwards, we loop through all patients, calculating the similarity between patient data and the discovered patterns by TRICLUSTER algorithm. Each discovered pattern (tricluster) consists of a set of biclusters, and a bicluster is composed of features. Therefore, we search for the patient's values for the corresponding bicluster's features and time point. Then, a vector with size $|J|$ is returned, where $|J|$ is the number of features. Once each slice of tricluster contains a set of values, we calculate the similarity coefficient between the patient data and the values from the pattern. If the vector has dimension 2, we use the cosine similarity as the similarity measure. Otherwise, if the dimension is greater than 2, the Pearson correlation is applied. Finally, we compute the mean of all similarity coefficients.

Algorithm .1: Correlation calculus

```
$Initialization : corr\_set =$; call
  correlations($data3d, X, T, corr\_set$);
begin
    foreach $patient \in X$ do
        foreach $tricluster \in T$ do
            foreach $bicluster \in tricluster$ do
                t = get_timepoint(bicluster);
                y = get_features(bicluster);
                data = data3d[patient][y][t];
                total_corr = 0;
                foreach $observation \in bicluster$ do
                    if len(y) == 2 then
                        corr = cosine_similarity(data, observation);
                    else
                        corr = pearson_correlation(data, observation);
                    total_corr = total_corr + corr;
                total_corr = total_corr / len(bicluster);
                corr_set[patient][tricluster][bicluster] = total_corr
    return corr_set
```

An illustrative dataset produced by this approach is presented in the Table I.

Patients	Corr $tri1$ $bi1$	$\dots$	Corr $triM$ biN
1	$\rho_{1,1,1}$	$\cdots$	$\rho_{1,M,N}$
2	$\rho_{2,1,1}$	$\cdots$	$\rho_{2,M,N}$
3	$\rho_{3,1,1}$	$\cdots$	$\rho_{1,M,N}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$

TABLE I: Tricluster-based approach: input data format

C. *Associative classification of MTS*

As introduced, we aim to create a superior approach to MTS classification. In order to assess our methodology, we compared three major approaches. The first one represents the approach of linear regression. The second discriminate all time points for each feature. Finally, the third contains our approach using the Correlations on the found triclusters. Besides, the combination of different approaches was also tried to exploit their advantages in the proposed 3D dataset classification model.

The Tricluster-based approach, which is present in this thesis, fits in the type of *Associative Time series classification (pattern-centric classifier)*, described in III-B. In the first phase, temporal patterns (triclusters) are detected, then a tabular dataset is created according to the similarity metric (Cosine similarity and Pearson Correlation). In Figure 3 is represented an comprehensive view of the proposed associative classification approach.

In order to assess the performance of our approaches, some classic classifiers have been applied such as Naive Bayes classifier **(NB)**, Support Vector Machines **(SVM)**, K-Nearest Neighbour **(KNN)**, Decision Trees **(DT)** and Extreme Gradient Boosting **(XGBoost)**.

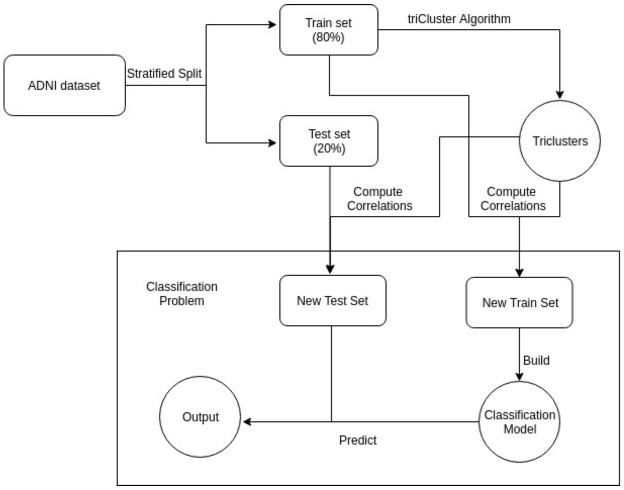


Fig. 3: Comprehensive view of the proposed classification process flow

D. *Hyperparameter optimization*

All classifiers were subject to hyperparameter optimization. The application of this process aims to discover a set of optimal hyperparameters for a learning algorithm. Each set of optimal parameters depends on the problem.

As a result, for each setting, a set parameters that minimise a established loss function is fixed/encountered. It is important to point out that hyperparameters tuning should be applied in a validation set, using cross-validation, avoiding overfitting.

Bayesian optimization. This approach ensembles a probabilistic model of the objective function mapping from hyperparameter values to the objective evaluated on a validation set. This approach seeks to obtain as much information as possible from the loss function testing a range of optimal parameters, which minimizes an objective function.

With this approach, we tried to find the "balance between exploration (hyperparameters for which the outcome is most uncertain) and exploitation (hyperparameters expected close to the optimum)" [23]. Finally, we intend to restrict evaluations of poor hyperparameter choices .

Throughout our implementation, we applied hyperparameter optimization on all classifiers. In each case, we split our dataset into a train set, a validation set and a test set, which the size of the validation set is the same as the test set. The hyperparameter optimization is done at the validation set in order to avoid overfitting. Bayesian optimization was the approach where we reach the best trade-off of performance and computational complexity.

Key concepts associated with the target Bayesian Optimization approach are:

1) *Objective Function*: Metric which we want to minimize. In our project, we used $1 - accuracy$ in the validation set;
2) *Domain Space*: Spectrum of parameters that will be search over;
3) *Optimization algorithm*: The method which is used to build the probabilistic model. In our work, we applied TPE [24]. TPE algorithm constructs the probability model from the past results and determines the next set of hyperparameters to evaluate the objective function;
4) *Result history*: The outcomes for each combination of parameterizations are stored in order to use the best one in the test set.

Hyperparameter domains The hyperparameter domains for each classifier are:

1) *NB:* no parameters were varied, a Gaussian (multinomial) assumption were considered on numeric (categoric) attributes;
2) *SVM*: i) type of kernel (linear and radial basis function kernel); ii) cost $C \in [1.0, 1e5]$; iii) gamma $\gamma \in [1e - 15, 1.0]$.
3) *KNN*; i) number of neighbors $k \in [3, 30]$;
4) *DT*: i) maximum depth $d \in \{3, 4, 5, 6, 7\}$;
5) *XGBoost*: i) maximum depth $d \in [3, 50]$, ii) learning rate $\alpha \in [1e - 3, 1.0]$, iii) number of estimators $N_e \in [5, 200]$; iv) number of leaves $N_l \in [3, 50]$.

V. Results

Our 3D dataset is composed of 1362 individuals, 85 features (14 temporal features) and 5 time points. The corresponding number of measurements for each class is described in Table II.

	CN:CN	CN:MCI	MCI:MCI	MCI:AD	AD:AD
Number	426	17	650	64	205

TABLE II: Number of measurements

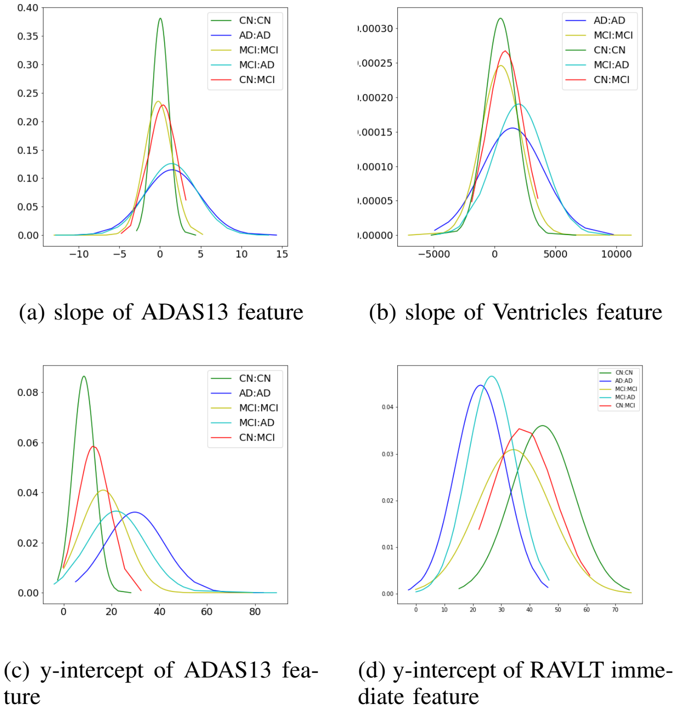
A. *Regression Coefficients*

As referred in subsection IV-B, we calculated a linear regression through 5-time points of each feature, and subsequently, we analyse the regression coefficients for each different stage of the disease. In this context, we intend to demonstrate that the regression coefficients differ according to the patient progression profile. In addition, we applied the D'Agostino's K-squared normality test, ensuring that the sample of coefficients is well-modelled by a Guassian distribution.

The respective p-values are 0.07 and 0.24 for the slopes of "ADAS13" and "Ventricles" features. Besides, the y-intercepts of "ADAS13" and "RAVLT_immediate" features obtained 0.08 and 0.22, respectively.

Since all p-values are above of 0.05, we cannot reject the hypothesis that the sample is well-modelled by a normal distribution. The respective plots are shown in Figures 4

First, Figures 4a and 4b represent the normal distributions for each class of linear regression slope of the following features: "ADAS13" and "Ventricles", respectively. Looking at the two plots, we can note that the representations of the normal distributions for each class are distinct. The mean values of the slope present slight differences compared with people in an advanced stage of the disease. Concerning the standard deviation, it is noticeable that the values increase with disease progression, where the curve of healthy people is narrow compared with more advanced stages of AD.



(a) slope of ADAS13 feature

(b) slope of Ventricles feature

(c) y-intercept of ADAS13 feature

(d) y-intercept of RAVLT immediate feature

Fig. 4

Second, Figures 4c and 4d display the normal distributions for each class of linear regression y-intercepts of the following features: "ADAS13" and "RAVLT_immediate". In figures 4c, we can conclude that healthy patients have lower mean and lower standard deviations, compared with patients in an advanced stage of the AD. Furthermore, 4d shows that the mean value of healthy people is higher compared with dementia profiles.

These illustrations confirm that linear regression coefficients have the capacity to discriminative different time series, highlighting the relevance of the regression-based approach as a MTS classifier.

B. Classification Problem

This sections presents the results of assessing the different MTS classification approaches. We performed stratified 10-fold cross-validation, guaranteeing that each fold is a good representative of the whole. Then, the values presented in tables below contain the averages of cross validation results with the respective standard deviation.

The implemented approaches are as follow: tricluster-based approach (T), simple approach (AT) and regression-based approach (LR). The combination of approaches is also displayed. Furthermore, we also present the classification metrics for two strategies: 1) building the model with only a subset of temporal features (14 features), and 2) integrating all temporal and static features (85 features), applying the mean substitution for the missing values, in the same tabular dataset.

Tables III and IV gather the performance of different classifiers (NB, SVM, KNN, DT, XGBoost) with three different approaches – tricluster-based features, simple method (all timepoints), and regression-based features. Table III attempts to do a fair evaluation, assessing the performance on the subset of 14 features. These features have a moderate to high variability of values and ratio of missings below 30 p.p.. The reason for this imposition is the lack of stability of TRICLUSTER algorithm to handle with missing data. In contrast with table III, table IV compares the Tricluster-based approach against the peer approaches on the complete set of features (85 features).

Approaches	**NB**	**SVM**	**KNN**	**DT**	**XGBoost**
T (14 features)	0.32 ± 0.04	$\mathbf{0.73 \pm 0.03}$	$\mathbf{0.68 \pm 0.04}$	$\mathbf{0.82 \pm 0.04}$	$\mathbf{0.86 \pm 0.04}$
AT (14 features)	$\mathbf{0.55 \pm 0.07}$	0.46 ± 0.05	0.48 ± 0.08	0.78 ± 0.06	0.85 ± 0.04
LR (14 features)	0.50 ± 0.04	0.42 ± 0.02	0.38 ± 0.05	0.76 ± 0.03	0.81 ± 0.03
T + LR (14 features)	$\mathbf{0.55 \pm 0.04}$	0.41 ± 0.05	0.39 ± 0.05	0.78 ± 0.03	0.85 ± 0.04
T + AT (14 features)	$\mathbf{0.55 \pm 0.08}$	0.46 ± 0.05	0.48 ± 0.08	0.81 ± 0.05	0.87 ± 0.05
T + LR + AT (14 features)	0.53 ± 0.07	$\mathbf{0.49 \pm 0.09}$	$\mathbf{0.49 \pm 0.07}$	$\mathbf{0.83 \pm 0.04}$	$\mathbf{0.88 \pm 0.04}$

TABLE III: Accuracy (14 features)

There are five major observations can be derived from results in Table III: 1) looking at each approach isolated, tricluster-based approach shows the highest performance for most classifiers. In particular, it shows an improvement 27 p.p. with SVM against the peer approaches (all timepoints and regression-based features) and an improvement of 20 p.p. with KNN; 2) the performance with NB decreases with tricluster-based features duo to the high dimensionality of the data space; 3) moderate improvements are observed with ensemble learning algorithms such as XGBoost; 4) combining features with DT and XGBoost contributes to a slight increase in performance, and the performance is achieved when combining tricluster-, simple- and regression-based features; 5) combining features has a deteriorated effect on the performance achieved by SVM and KNN.

Approaches	NB	SVM	KNN	DT	XGBoost
T (14 features)	0.32 ± 0.04	$\mathbf{0.73 \pm 0.03}$	$\mathbf{0.68 \pm 0.04}$	0.82 ± 0.04	0.86 ± 0.04
AT (85 features)	0.51 ± 0.07	0.43 ± 0.04	0.46 ± 0.07	$\mathbf{0.84 \pm 0.03}$	$\mathbf{0.89 \pm 0.03}$
LR (85 features)	$\mathbf{0.53 \pm 0.04}$	0.42 ± 0.02	0.39 ± 0.04	0.83 ± 0.04	0.87 ± 0.03
T + LR (85 features)	$\mathbf{0.54 \pm 0.04}$	0.47 ± 0.00	0.39 ± 0.04	0.83 ± 0.03	0.89 ± 0.02
T + AT (85 features)	0.50 ± 0.07	0.42 ± 0.04	$\mathbf{0.46 \pm 0.07}$	$\mathbf{0.85 \pm 0.03}$	0.90 ± 0.03
T + LR + AT (85 features)	0.50 ± 0.06	0.57 ± 0.06	0.44 ± 0.05	$\mathbf{0.85 \pm 0.02}$	$\mathbf{0.91 \pm 0.03}$

TABLE IV: Accuracy (85 features)

In addition, three significant observations can be obtained from table IV: 1) tricluster-approach is still the top performer with SVM (+20 pp) and KNN (+22 pp), even considering the original set of features for the simple approach and regression-based features; 2) given the information value of the excluded features, DT and XGBoost show a slightly higher performance with regression-based features; 3) summing up, combining tricluster-based features shows an unarguable improvement in the performance of MTS classifiers.

Furthermore, we chose the Cohen Kappa statistic to assess the performance of the classifiers. This metric is relevant, especially on imbalanced data such as ours.

Approaches	NB	SVM	KNN	DT	XGBoost
T (14 features)	0.19 ± 0.04	$\mathbf{0.60 \pm 0.06}$	$\mathbf{0.54 \pm 0.06}$	$\mathbf{0.74 \pm 0.05}$	$\mathbf{0.79 \pm 0.04}$
AT (14 features)	$\mathbf{0.40 \pm 0.10}$	0.16 ± 0.08	0.25 ± 0.12	0.67 ± 0.09	0.78 ± 0.05
LR (14 features)	0.30 ± 0.05	0.03 ± 0.05	0.11 ± 0.07	0.65 ± 0.05	0.73 ± 0.04
T + LR (14 features)	0.37 ± 0.07	0.04 ± 0.04	0.12 ± 0.07	0.70 ± 0.06	0.79 ± 0.04
T + AT (14 features)	$\mathbf{0.41 \pm 0.11}$	0.16 ± 0.08	$\mathbf{0.25 \pm 0.12}$	0.74 ± 0.07	0.80 ± 0.05
T + LR + AT (14 features)	0.39 ± 0.09	$\mathbf{0.24 \pm 0.15}$	$\mathbf{0.25 \pm 0.10}$	$\mathbf{0.74 \pm 0.07}$	$\mathbf{0.81 \pm 0.04}$

TABLE V: Cohen's kappa (14 features)

Regarding the Table V, the final observations are: 1) tricluster-based approach shows a highest performance for most classifiers, except for NB; 2) the performance with NB decreases with tricluster-based features duo to the high dimensionality of the data space, as referenced above; 3) combining features with DT and XGBoost contributes to a slight increase of performance. For all classifiers except NB, the best performance is achieved when combining tricluster-, original- and regression-based features.

C. *Three-dimensional patterns*

In this section, the discriminative power of an illustrative set of found triclusters is discussed. For this end, we used the feature importance of XGBoost to verify the triclusters which had more influence on model construction.

In Figure 5, we can observe the features that were most selected in the construction of decision trees by XGBoost.

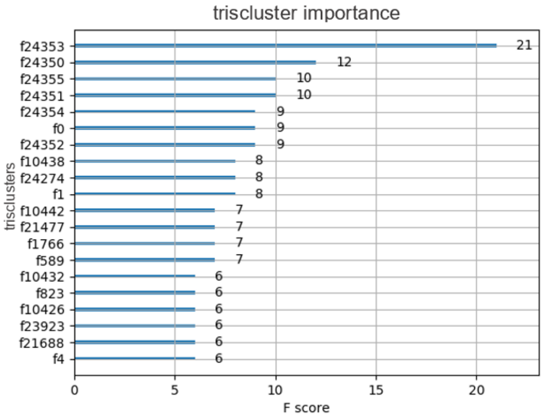


Fig. 5: XGBoost model: tricluster importance

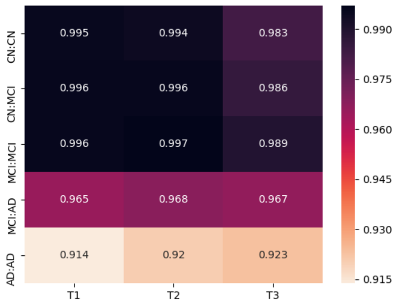


Fig. 6: Mean correlation between each target class and triclusters 24350 to 24355

Regarding the Figure 5, we can conclude that the triclusters 24350 to 24355 present a major importance on model construction. Then, we further analysed the composition of these tri-dimensional patterns, corresponding a 3 triclusters (6 biclusters) with the same composition. The composition is: $J = \{$'CDRSB', 'RAVLT_immediate'$\}$ and $K = \{t_0, t_4\}$, containing only MCI:MCI datapoints. In the following Figure 6, the mean correlations of the patients in each class with the three triclusters are represented as a heatmap.

Concerning the Figure 6, we can rank the classes according to the correlation with the triclusters, as: 1) MCI:MCI, 2) CN:MCI, 3) CN:CN, 4) MCI:AD and 5) AD:AD. As expected, the class MCI:MCI has mean correlations closest to 1, because the triclusters are just formed by individuals in this stage of disease. Nonetheless, it is possible to verify that the classes CN:MCI and CN:CN have also a strong correlation with these 3 triclusters. This representation helps us to prove that the found triclusters and the applied similarity metric can distinguish individuals in different stages of the disease.

As a final thought, we hypothesize that the cumulative discriminative contributions from the various triclusters (each with mild/moderate discriminative power) is potentially the underlying cause behind the promising results of the Tricluster-based approach for MTS classification.

VI. Conclusion

This thesis proposed a new approach for MTS data analysis using the clinical trials from ADNI data, with a focus on the progression of Alzheimer's disease. Its aim is to explore discriminative patterns observed along the different stages of AD and assess, whether they can enhance the classification of individuals.

This work focused on a 3D dataset consisting of 1362 patients, 85 features (14 temporal features) and 5-time points.

With the purpose of finding non-trivial patterns throughout the MTS, we applied the TRICLUSTER algorithm, which is up to date a state-of-the-art triclustering algorithm of reference. Then, we created an effective way of using temporal patterns in the associative classification model, introducing the Correlation approach for the discriminative power calculus. This approach combines both the cosine similarity and Pearson correlation to assess the likelihood of a given individual displaying a specific temporal pattern of interest.

Therefore, with the purpose of setting a benchmark, making a fair judgement of the Tricluster-based approach potentialities, we introduced two baselines. First, the ”simple approach” maps all time points into a tabular dataset, where the 3D dataset is transformed into a matrix. Second, we presented the regression-based approach, making use of a temporal abstraction to capture the progression of values along time. For the second baseline, it was implemented a normality test to assess the capability of describing the regression parameters by Gaussian Distribution and whether the distribution of values per parameter differed between classes. Also, we performed several types of hyperparameter optimisation, where Bayesian optimisation achieved the best time versus performance trade-off.

For the proposed work, our novel approach outperforms the two baselines, emphasising the discriminative power of the triclusters. Additionally, an ensemble approach, combining the features produced by all the approaches, was proposed. Results show that it is able to improve the performance of most classifiers, including XGBoost, over the isolated approaches. As a final thought, we want to emphasise that the promising results gathered on this work, demonstrate the key role of three-dimensional patterns in MTS data classification.

References

[1] “2018 alzheimer’s disease facts and figures,” *Alzheimer’s Dementia*, vol. 14, no. 3, pp. 367 – 429, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1552526018300414

[2] World Health Organization, *Towards a dementia plan: a WHO guide*, 2017. [Online]. Available: http://www.who.int/mental_health/neurology/dementia/en/

[3] A. Saeb, S. David, M. Rafiullah, and K. Al-Rubeaan, *Classification Techniques and Data Mining Tools Used in Medical Bioinformatics*, 11 2018, pp. 105–126.

[4] A. Mueen, E. Keogh, and N. Young, “Logical-shapelets: An expressive primitive for time series classification,” in *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’11. New York, NY, USA: ACM, 2011, pp. 1154–1162. [Online]. Available: http://doi.acm.org/10.1145/2020408.2020587

[5] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, “Time series classification using multi-channels deep convolutional neural networks,” in *Web-Age Information Management*, F. Li, G. Li, S.-w. Hwang, B. Yao, and Z. Zhang, Eds. Cham: Springer International Publishing, 2014, pp. 298–310.

[6] A. J. Saykin, L. Shen, X. Yao, S. Kim, K. Nho, S. L. Risacher, V. K. Ramanan, T. M. Foroud, K. M. Faber, N. Sarwar, L. M. Munsie, X. Hu, H. D. Soares, S. G. Potkin, P. M. Thompson, J. S. Kauwe, R. Kaddurah-Daouk, R. C. Green, A. W. Toga, and M. W. Weiner, “Genetic studies of quantitative MCI and AD phenotypes in ADNI: Progress, opportunities, and plans,” *Alzheimer’s and Dementia*, vol. 11, no. 7, pp. 792–814, 2015.

[7] M. J. Zaki and W. M. Jr., *Data Mining and Analysis: Fundamental Concepts and Algorithms*.

[8] R. Henriques and S. C . Madeira, “Triclustering algorithms for three-dimensional data analysis: A comprehensive survey,” *ACM Computing Surveys*, vol. 51, pp. 1–43, 09 2018.

[9] A. Bhar, M. Haubrock, A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and E. Wingender, “-trimax: Extracting triclusters and analysing coregulation in time series gene expression data,” in *WABI*, 2012.

[10] A. Li and D. Tuck, “An effective tri-clustering algorithm combining expression data with gene regulation information.” *Gene regulation and systems biology*, vol. 3, pp. 49–64, 2009.

[11] H. Jiang, S. Zhou, J. Guan, and Y. Zheng, “gTRICLUSTER : A More General and Effective 3D Clustering Algorithm for Gene-Sample-Time Microarray Data,” *BioDM*, no. 60373019, pp. 48–59, 2006.

[12] L. Zhao and M. J. Zaki, “Tricluster,” *Proceedings of the 2005 ACM SIGMOD international conference on Management of data - SIGMOD ’05*, no. September, p. 694, 2005. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1066157.1066236

[13] Y. C. Liu, C. H. Lee, W. C. Chen, J. W. Shin, H. H. Hsu, and V. S. Tseng, “A novel method for mining temporally dependent association rules in three-dimensional microarray datasets,” *ICS 2010 - International Computer Symposium*, pp. 759–764, 2010.

[14] I. Batal, L. Sacchi, R. Bellazzi, and M. Hauskrecht, “Multivariate time series classification with temporal abstractions,” in *FLAIRS Conference*, 2009.

[15] P. Esling and C. Agón, “Time-series data mining,” *ACM Comput. Surv.*, vol. 45, pp. 12:1–12:34, 2012.

[16] A. Abanda, U. Mori, and J. A. Lozano, “A review on distance based time series classification,” *Data Mining and Knowledge Discovery*, vol. 33, no. 2, pp. 378–412, Mar 2019. [Online]. Available: https://doi.org/10.1007/s10618-018-0596-4

[17] P. Geurts, “Pattern extraction for time series classification,” in *Principles of Data Mining and Knowledge Discovery*, L. De Raedt and A. Siebes, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 115–127.

[18] M. Hüsken and P. Stagge, “Recurrent neural networks for time series classification,” *Neurocomputing*, vol. 50, pp. 223 – 235, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231201007068

[19] V. Pavlovic, B. J. Frey, and T. S. Huang, “Time-series classification using mixed-state dynamic bayesian networks,” in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, vol. 2, June 1999, pp. 609–615 Vol. 2.

[20] B. Esmael, A. Arnaout, R. K. Fruhwirth, and G. Thonhauser, “Improving time series classification using hidden markov models,” in *2012 12th International Conference on Hybrid Intelligent Systems (HIS)*, Dec 2012, pp. 502–507.

[21] H. Kang, “The prevention and handling of the missing data,” *Korean journal of anesthesiology*, vol. 64, pp. 402–6, 05 2013.

[22] R. B. D’agostino, A. Belanger, and R. B. D. Jr., “A suggestion for using powerful and informative tests of normality,” *The American Statistician*, vol. 44, no. 4, pp. 316–321, 1990. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00031305.1990.10475751

[23] Y. Xia, C. Liu, Y. Li, and N. Liu, “A boosted decision tree approach using bayesian hyper-parameter optimization for credit scoring,” *Expert Systems with Applications*, vol. 78, pp. 225 – 241, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417417301008

[24] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, “Algorithms for hyper-parameter optimization,” in *Proceedings of the 24th International Conference on Neural Information Processing Systems*, ser. NIPS’11. USA: Curran Associates Inc., 2011, pp. 2546–2554. [Online]. Available: http://dl.acm.org/citation.cfm?id=2986459.2986743