Classification of clinical time series:

A case study in patients with Multiple Sclerosis

André V. Carreiro

Instituto Superior Técnico

Universidade Técnica de Lisboa

Lisbon, Portugal

nava_carreiro@hotmail.com

***Abstract* — The constant drive towards a more personalized medicine in the last years led to the arrival of temporal gene expression analyses. Due to the consideration of a temporal aspect, this kind of analyses represents a great advantage to better understand disease progression and treatment results at a molecular level. Nevertheless, several problems accompany studies of this kind, with the sample size limitation being one of the most relevant. This limitation is patent in two ways: in the number of objects (in this case, patients), and in the number of measured time points. In this work, the data used were multiple gene expression time series, used to classify the response of multiple sclerosis patients to the standard treatment with Interferon-β, to which nearly half of the patients reveal a negative response. Therefore, obtaining a highly predictive model of a patients' response would definitely improve the quality of life, avoiding useless and possibly harmful therapies for the non-responder group.**

In this context, several new strategies for time series classification are proposed, based on a biclustering technique. These are applied to the classification of the Interferon-β response by multiple sclerosis patients from a dataset analyzed over the last decade. Although our classification methods do not outperform the ones proposed for this same dataset, it is worth noting that some of the developed strategies reveal important potentialities that should be further explored, either with other clinical time series data, or even in other classification problems, in general.

Keywords - time series; biclustering; bioinformatics; multiple sclerosis; IFN-β; data mining

I. BACKGROUND AND STATE OF THE ART

A. Multiple Sclerosis: Mechanisms and Treatment

MS can be defined as a chronic inflammatory disease, characterized by a demyelinating disorder of the central nervous system (CNS) [1]. Although its etiology remains, to date, still far from total understanding, the interrelation of both genetic and environmental factors is believed to be of crucial importance to the development of MS. A major factor to be considered in this regard is the phenotypic heterogeneity in MS, where different pathologic patterns may indicate differences in the pathogenic mechanisms [1]. Moreover, the search for single candidate genes that could account for the disease development is still unfruitful. The main conclusion is that MS is a genetically complex disease, and it is not possible to select single genes to explain a person's susceptibility, since it might be a result of the interaction of several altered genes [1].

Consequent to the heterogeneity of the disease, the treatment response, even for one stage of MS only (relapsing remitting MS), presents a high variability, suggesting different responses at the molecular level, leading to diverse clinical outcomes as the inhibition of the CNS inflammation [2]. Nevertheless, the treatment of RR-MS patients has routinely been carried with the use of recombinant human interferon b (rIFN-b) [3]. However, up to half the patients show no benefits from this treatment, and negative side effects such as flu-like symptoms and tissue damage have to be considered [4]. In this context, the main goal of a time-course profiling of the treatment response of MS patients rests, as can be anticipated, in the possibility of accurately predicting a given patient's response, avoiding useless and possibly harmful treatments.

B. Temporal gene expression analysis

At a given instant of time, only a small fraction of the organism's genome is, indeed, expressed. In the last decade, several techniques of evaluating genes expression became available, such as the gene arrays, which measure, instantly, the expression level of up to thousands of genes (possibly, all genes in a genome). Basically, the intent with these experiments is to observe the hybridization of the mRNA molecules, the so-called targets, to some predefined probes. These experiments can be classified according to its type or to the profiled organism, and the probes can be cDNA or short oligonucleotides [5].

Gene-expression experiments would, until more recently, be limited to a static analysis, in which only a snapshot of the gene expression for a set of samples was available. However, the last years have witnessed the arrival and evolution of time-course gene expression experiments. With the possibility to perform a temporal analysis of gene expression, some questions of great importance arise, such as the problem of identifying coherent responses, which can distinctly characterize various classes of objects, either conditions or patients. In other words, this problem lies in the identification of combinations of up- or down-regulated genes, or, more interestingly, genes with coherent expression profiles, then used for disease classification.

Computational challenges and methods of analysis

The computational challenges faced when analyzing time series expression data can be divided into four levels [5]:

a) Experimental design

This level consists mainly in the previous establishment of the required number of microarrays and representative probes for a determined gene sequence, aiming to minimize the possibility of cross-hybridization [6].

Several different limitations, of technological or practical order, limit the sample size. On one hand, the number of time points that can be measured is very restricted., and equally or even more significantly, the number of biological and technical replicates is also an important problem [7].

Regarding the noisy data issue, unless there is some knowledge about the implicit concept generating the data, the detection and distinction of noise is a really difficult task to complete.

The determination of the sampling rates is also an important issue, to avoid both under- and oversampling. The other major challenge of the experimental design is concerned with the synchronization of the cells, that might be lost after some time [5].

b) Data analysis

In this level, the focus is on the individual gene. This can be easily stated in tasks like the study of the continuous evolution for each gene, gene alignment, identification of differential expression, and coherent expression patterns, from different time-course expression experiments [5]. The presence of noise in the data is inevitable, and a solution to deal with this issue and small number of replicates must go past the simple methods of interpolation of individual genes, which lead to inadequate estimates [5]. Another major challenge in the data analysis is the variability of the timing of the biological processes, since it differs between organisms, genetic variants and environmental circumstances.

c) Pattern recognition

Point-wise distance-based clustering methods

With these methods, the clustering is achieved by determining the distance between two samples, and creating clusters with samples that fall within a certain threshold. That distance, or similarity, can be measured with norm-based distances and combination of correlation metrics [7], including methods such as k-means, self-organizing maps (SOM) and hierarchical clustering.

Model-based clustering methods

In this type of clustering methods, instead of focusing the similarity on the data, *per se*, the similarity measure is based on an unknown model built to describe the data. The goal here is the identification of a mixture model, given by the appropriate combination of base functions, capable (most as possible) of explaining the data. To our knowledge, the currently most promising clustering method for time series expression datasets is based on Hidden Markov Models (HMMs). These allow to overcome the lack of consideration for the temporal nature of the data, leading to a more efficient clustering [8]. A good introduction to HMMs and its properties can be found in [9].

Feature-based clustering methods

The goal of this sort of clustering methods is the identification of prominent features from the expression profiles, analyzing local or global consistencies in transformed data [7], rather than using the aforementioned quantifiable metrics. This results in a higher flexibility, minimizing the influence of noise and uncertainties associated with the mRNA expression quantification [7].

Biclustering

A bicluster can be defined as a subset of genes that display an analogous expression pattern for a subset of conditions or time points (in the time series analysis) [10]. The key advantage presented by this method is the opportunity to unravel processes which are not active during all the time points or across all conditions, but only for a smaller set.

Most versions of the biclustering problem are NP-hard [11]. Nonetheless, in the case of time series expression data the biclustering problem can be restricted to finding biclusters with coherent patterns and contiguous time points. This restriction leads to a tractable problem. In this work, we use CCC-Biclustering [11], which finds all maximal contiguous column coherent (CCC) biclusters (subsets of genes with coherent expression patterns in contiguous subsets of time points) by analyzing a discretized version of the expression matrix using efficient string processing techniques based on suffix trees. The biclustering-based classifiers proposed in this work use CCC-Biclusters as the class discriminative features.

d) Networks

One can state that this is a higher level of the general time-course expression experiments, because the attention is on the genes interaction and different systems in the whole cell, trying to model them, either descriptively or predictably. The first challenge arising at this level, both for static and time series experiments, comes from the necessary combination of data from different biological sources, which includes protein-DNA and protein-protein interactions, as well as prior knowledge on the expression data [5].

C. Related work: classification of clinical time series

Baranzini et al. [3] collected a dataset (Section 3) containing the profiling of MS patients subjected to IFN-b therapy. These authors proposed a quadratic analysis based integrated Bayesian inference system (IBIS) to analyze it. They chose the best discriminative triplets of genes, obtaining a prediction accuracy up to 86% for a gene triplet consisting of Caspase 2, Caspase 10 and FLIP. We note, however, that in this work only the first time point was considered. Lin et al. [8] proposed a new classification method, based on Hidden Markov Models (HMMs) with discriminative learning (using both positive and negative examples). In this work, the analysis was preceded by a feature selection step, to eliminate the least discriminative genes.

The main results of applying this method to the MS dataset for two to seven time points were a prediction accuracy of up to 88%, and most importantly, the consideration and identification of patient-specific response rates. Finally, Costa et al. [4] introduced the concept of constrained mixture estimation of HMMs and applied it to the MS dataset. The constraints were positive when two patients were forced to be associated in the same group, or negative when they were not allowed to be grouped together. A preprocessing feature selection step was also performed. The main results include a prediction accuracy over 90% and the possibility of subgroup classification (two subgroups of good responders). This method also suggested the existence of one mislabeled patient, which was confirmed by Baranzini et al. [3].

II. Methods

A. Dataset description and preprocessing

The dataset used as case study in this work was collected by Baranzini et al. [3]. Fifty two patients with relapsing-remitting (RR) MS were followed for a minimum of two years after the treatment initiation. After that time, patients were classified according to their response to the treatment, as good or bad responders. Thirty two patients were considered good responders, while the remaining twenty were classified as bad responders to IFN-b therapy. Seventy genes were pre-selected based on biological criteria, and their expression profile was measured in seven time points, using one-step kinetic reverse transcription PCR [3].

In order to apply CCC-Biclustering [11], as part of the proposed biclustering-based classifiers, we normalized and discretized the expression data. The discretization was performed by computing variations between time points as performed by Madeira et al. [11], thus resulting in patterns of temporal gene expression evolution with three symbols: decrease (D), no change (N) and increase (U). However, in this work genes with missing values are not discarded *apriori*. Instead, an adapted version of CCC-Biclustering, able to cope with missing values directly, is used. In the case of standard classifiers, not able to deal with missing values directly, these were filled with the average of the closest neighboring values, after data normalization.

B. Classification

In this subsection we present the new biclustering-based classifiers we developed in this thesis.

1) Biclustering-based k – Nearest Neighbors

The kNN algorithm is a simple supervised learning method, whose goal is to classify an object based on the k closest training instances. The k parameter is a positive integer, usually small, chosen empirically with the help of some cross validation schemes, for example. In order to favor the best scoring instances, a distance-weighted algorithm can be used. The weights can be a function of their rank (with a weight of 1/d, d being the rank or the distance to the test object). Algorithm 1 shows the biclustering-based kNN algorithm used to classify patient responses.

Algorithm 1: Biclustering-based kNN

```
Input : Score matrix between patients: S
Output: predictedClass
foreach test patient do
    build list with k highest scoring train patients → kPatients
    from kPatients, separate the scores for each class: scores0 and scores1
    predictedClass = 0
    if sum(scores1) > sum(scores0) then
        predictedClass = 1
```

Computation of the score matrix between patients

The matrix that represents the relationship between test and train patients, from where the k most similar train patients are selected to classify each test patient, is the score matrix (the higher the score, the higher the degree of similarity between the patients). It has as many rows as the number of train patients, and a number of columns equal to the number of test patients. To reduce the effects of unbalanced data, as is the case with this dataset, a penalty/weight can be included in the computation of this matrix, altering the actual score between the two patients.

a) Biclusters similarities

In this case, we have a previous matrix which represents the similarities between the biclusters of each pair of test and train patients. The entry (i,j) of this matrix represents the degree of similarity between a bicluster B_i from the set of biclusters of a test patient, and a bicluster B_j from the set of biclusters of a train patient. This similarity can be computed from the fraction of common elements of the two biclusters in comparison, using an adapted version of the Jaccard Index, used in [11]:

- in the two dimensions (genes and time points), with or without the expression pattern information.
- just in the genes dimension, again with or without the information of the bicluster profile, i.e. the symbols of the discretized matrix that represent the temporal evolution of the gene expression.

However, it is necessary to transform this measure of similarity between two patients in a single score value, to proceeed with kNN classification, which is not possible with the previous matrix. This transformation can be carried in the following manner, although there are other possibilities: for each test bicluster B_i, find the best similarity along the train biclusters B_j. The score that represents the relationship between the two patients S(P_test,P_train), is the average (of the maximum) similarity along the set of test biclusters:

$$S(P_{test}, P_{train}) = \frac{\sum_{i=1}^{\#B_{test}} \max(Sim(B_i, B_j), j \in \{1, \dots, \#B_{train}\})}{\#B_{test}} \quad (1)$$

where #B_test and #B_train represent the number of biclusters of, respectively, the test and train patient.

Filter non-discriminative biclusters based on similarities

The importance of a feature selection was already pointed out in the related work section (I.C), since better prediction accuracies in the classification results were obtained for a reduced set of most discriminative genes. Keeping this concept in mind, a new filter is proposed to eliminate features, which are in this case biclusters, instead of only genes, that discriminate poorly the two classes. The decision to eliminate a bicluster can be based in Definition 1.

Definition 1 – Discriminative bicluster

A bicluster is discriminative for a class c, and so should be maintained in the feature space, if and only if the proportion of similar biclusters (above a similarity threshold) of the class c is greater than a predefined class proportion threshold.

b) *Profile similarities*

Another strategy developed to compute the score matrix between the test patients and the training set, relies on the fact that each bicluster is represented by a pattern of symbols, a profile, result of the discretization of the normalized matrix of gene expression values, and representative of that gene's expression evolution along the time points.

The conditions that have to be met in order to state that a certain profile is shared between two patients can be adjusted, such as the minimum number of common genes and/or time points. This means that besides representing the same expression pattern, a shared profile has to represent biclusters that have the required minimum number of genes and time points in common. Using this concept, the score matrix between patients is computed, in which an entry (i,j) represents the number of profiles shared between train patient i and test patient j.

Instead of the sum of shared profiles between patients, the entry (i,j) of the score matrix can be computed with a polynomial kernel [12](generally a quadratic kernel is used). With this measure, the patients with a greater number of biclusters are penalized, since a higher number of profile matches could be due to random events.

Filter non-discriminative biclusters based on profiles

In this situation, a given profile is kept in the filtered set, if and only if it contributes more to the discrimination than to the confusion between classes, that is, a profile in a train patient's set of profiles is maintained if and only if is shared by more patients of the same class than of the other class. A parameter of a minimum number of genes and/or time points shared can also be included and fine-tuned.

c) *Comparing discretized matrices*

Symbol pairing

To overcome the previous situation, we devised a new strategy that considers the nature of the comparison between symbols belonging to biclusters (genes' expression temporal variation). For that purpose, each element of the discretized matrix for one patient (with all or only a given number of biclusters, and considering that 'X' is the symbol used to represent the elements that do not belong to any of the selected biclusters) is compared to the corresponding element of the other patient's discretized matrix, considering the following symbol comparison possibilities: X-X, X-{U,N,D}, {U,N,D} mismatch, {U,N,D} match. The division of the different match possibilities has the goal of allowing a certain flexibility in the results, since one pairing might have a medium importance, like an X-{U,N,D} pair, while the most important one remains the perfect match (exception made to the X-X pair).

Note that if only the perfect match is to be considered, there is a simple method of comparison returning 1 if the elements are equal and 0 otherwise. The sum of such a binary matrix defines the score between the two patients.

Symbol pairing with time-lags

The last of these three strategies introduces a new consideration to the previous one: the possibility of time-lags in the gene expression. As one might expect, even when the same genes are involved in some mechanism for different patients, the expression evolution pattern for one patient might be delayed when compared to other's. This possibility should be taken into account, as it is a consequence of the patient-specific response rate, not considered in the methods proposed so far, and shown to be of particular importance in previous time-series expression studies [8].

In this approach, all the biclusters (or filtered ones) of the test patient are analyzed. A parameter for a maximum time-lag (instants to consider in the delay) is defined, and then, for each of the test biclusters, a comparison similar to the previous one, only considering the perfect matches, is then performed considering translations in the time points, from 0 (the original position) to the maximum time-lag, and its symmetric, allowing translations in both directions along the time axis.

The time-lag that returns the highest score is chosen, and the binary submatrix resulting from that specific comparison is written in a final matrix. The sum of this final matrix represents the score between the two patients, the entry (i, j) of the score matrix for the whole set of patients.

2) *Meta-profiles classification*

Having explored different strategies to combine biclustering and kNN classification, we now present a new classification approach following the biclusters computation. It is based on the mentioned fact that each bicluster has a pattern of temporal evolution in terms of gene expression, which is represented by a profile. A meta-profile represents a set of equivalent profiles. In this approach, described in Algorithm 2, the goal is to analyze if a given profile is shared between more patients of one of the classes. For example, if a train profile is shared only between good responders, then if a test patient shows an equivalent expression profile, the probability of this patient being a good responder increases. In this method, the class proportions for each profile of a test patient contribute for the patient classification, in a weighted-voting scheme. Due to the difference in the class distributions, a penalty can also be introduced here to soften the binary classification. Opposed to expected, the performed tests revealed that the best discriminative criterion was that the patients with more balanced class proportions were classified as good responders (class 1).

Algorithm 2: Meta-Profiles Classification

```
Input : Meta-Profiles space: vector with all computed biclusters profiles
Output: predictedClass
foreach meta-profile m do
    foreach train patient tp do
        TrainIndexes ← {}
        if meta-profile m belongs to set of profiles of tp then
            add tp to TrainIndexes
    compute meta-profile m classes proportions: Proportions0 and Proportions1
foreach test patient i do
    foreach test profile p do
        if p belongs to meta-profiles space then
            compute the sum of class proportions for all test profiles: sumProportions0 and sumProportions1
    predictedClass = 0
    if sumProportions1 x penalty < sumProportions0 then
        predictedClass = 1
```

3) *Meta-biclusters classification*

In this method, biclustering is used as a preprocessing step, to build a binary matrix which can then be fed to more standard classification techniques. The principle is based on meta-biclusters, which represent a set of similar biclusters. These are obtained by performing a hierarchical clustering in the bicluster space, for all patients, where the number of meta-biclusters is a user-defined parameter. The result of this clustering is used to build a binary matrix, where 1's represent the meta-biclusters representing at least one of the patient's biclusters.

4) *State of the art classifiers*

In order to obtain a term of comparison for the obtained results from the classification methods proposed in this thesis, the dataset was classified using state of the art classifiers, either from the original dataset, either from the result of the meta-biclusters method aforementioned. This was carried using the software package Weka (Waikato Environment for Knowledge Analysis), available in www.cs.waikato.ac.nz/ml/weka. It is open source software, issued under the GNU General Public License, and consists of a collection of machine learning algorithms for data mining tasks. We resorted to classifiers such as decision trees, kNN, Support Vector Machines (SVM), logistic regression, multilayer perceptron (MLP) and radial basis function (RBF) network [12].

C. *Evaluation*

We are dealing with a classification task to predict a given patient's response to a MS treatment. As such, the evaluation of any method must be based on the algorithm capability of predicting the responder class of a new MS patient, based on similar data than that used to train the classifier. For a finite dataset, the prediction accuracy is defined as the percentage of correctly predicted instances. Some problems are associated, however, as the assumption of equal misclassification costs and approximately uniform class distribution, which do not happen in most of the clinical classification problems (e.g., 95% of healthy individuals in a cancer case study). To overcome these issues, there are other possibilities in terms of evaluating a classifier's performance.

1) *Confusion matrix*

The confusion matrix for a classifier can provide more information about its performance [13].

TABLE I – Example of a confusion matrix for binary classification

		Predicted	
		negative	**positive**
Real	**negative**	**a** TN – true negatives	**b** FP – false positives
	positive	**c** FN – false negatives	**d** TP – true positives

From these values, we can compute the two most used ratios in classification performance evaluation:

- True positive (TP) rate = TP/(FN+TP), also called as the sensitivity of the classifier;
- False positive (FP) rate = FP/(TN+FP).

2) Receiver operating characteristics curve

Another possibility to analyze a classifier's performance is to visualize the two most informative values: TP rate and FP rate, by a curve, such as the Receiver Operating Characteristics (ROC) curve. This curve provides information about a classifier's performance for all misclassification costs, and all possible class ratios. It is a plot that relates the FP rate on the x-axis, and the TP rate on the y-axis [13]. When only a few (FP, TP) points are computed, an interpolation results in an approximate ROC curve.

3) Cross validation

The most usually chosen strategy to evaluate accuracy estimation is the k-fold cross validation (CV) scheme, where the dataset D is randomly partitioned into k mutually exclusive subsets, of equal size (or approximately): $D_1, \ldots D_k$. Then, for each fold (1 to k), the classifier is trained with the $k-1$ remaining subsets, and tested with the subset D_k. This is repeated k times, and the overall prediction accuracy estimate is the mean prediction accuracy for all k folds [14]. When k equals the number of instances in the dataset, it is called as Leave-one-out (LOO) CV, because one instance is left out to test the classifier, while all the others constitute the training set. Finally, if the class proportions in the original dataset are maintained in the subsets or folds, it is called a stratified cross validation [14]. In this work, we use both LOO and 5 repetitions of stratified 4-fold CV.

III. Experimental Results

A. Biclustering-based classifiers

Due to space limitations, an example of an obtained confusion matrix for the meta-profiles classification method is shown in TABLE II (sum criterion with penalty = 0.62), resulting in a prediction accuracy of 86.54%.

TABLE II - Confusion matrix obtained for meta-profiles classification (sum criterion with penalty = 0.62)

		Classified as	
		0	1
Real clas	0	17 (85.00%)	3 (15.00%)
	1	4 (12.50%)	28 (87.50%)

The approximate ROC curve built from tests with different values for the penalty used in the sum criterion in meta-profiles classification is shown, as an example, in Figure 1.

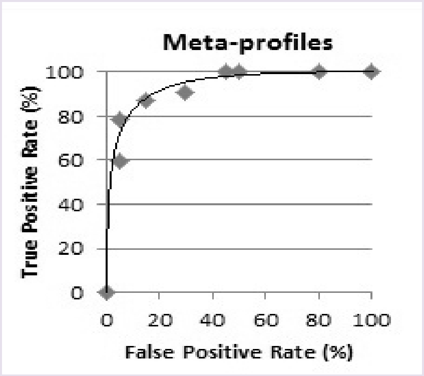

Figure 1 - Approximate ROC curve constructed with variations of the penalty in the sum criterion for meta-profiles classification.

We further note that, since this approach was the one that returned higher values of prediction accuracy, we performed new tests, reducing the set of biclusters to the 90% best (in terms of p-value, computed as in [11]). The obtained prediction accuracy increased, for some penalty values, up to 94%, revealing the importance of a feature selection step.

TABLE V summarizes the main results of prediction accuracies obtained for the proposed biclustering-based classifiers. It is important to refer the fact that the 61.54% of prediction accuracy obtained with the kNN classifier based on the comparison between elements of biclusters (TABLE V), corresponds to the classification of all patients as good responders.

In TABLE III are represented the best values of prediction accuracy obtained for the different state of the art classifiers (from the collection available in the software package Weka), applied on the original expression data.

Similarly, TABLE IV shows the best prediction accuracies obtained with the same collection of classifiers, this time applied on a discretized version of the expression data, to assess the influence of the discretization process previous to biclustering.

TABLE III - Summary of the main results (prediction accuracy) obtained from standard classifiers using the numeric dataset.

	Decision Tree	*KNN*	*SVM*	*Logistic Regression*	*RBF Network*	*MLP*
Pred. Acc. (LOO) (%)	71.15	86.54	92.31	80.77	88.46	86.54
Pred. Acc. (5 x 4 fold) (%)	70.77	82.31	85.00	80.38	83.85	86.15

TABLE IV - Summary of the main results (prediction accuracy) obtained from standard classifiers using a discretized version of the expression data.

	Decision Tree	*KNN*	*SVM*	*Logistic Regression*	*RBF Network*	*MLP*
Pred. Acc. (LOO) (%)	51.92	55.77	59.62	40.38	57.69	46.15
Pred. Acc. (5 x 4 fold) (%)	54.61	49.62	53.08	45.77	56.15	57.95

IV. DISCUSSION

It is important to emphasize some particular characteristics of this dataset, where beside a class unbalance, we find significant differences between what is shared between the patients of the two responder classes: good responders seem to share a large number of biclusters/profiles between them, and also with bad responders. On the contrary, bad responders, apparently, do not share sufficient information (biclusters/profiles) between them, decreasing the prediction accuracy for most of the developed strategies.

The most problem-specific biclustering-based strategy, for which the best results were obtained was the meta-profiles classification. As discussed before, the unanticipated criterion might be a result of the data particular characteristics, and should be further explored.

It was not possible to reproduce the results of the previous work on the MS dataset [4], due to serious difficulties in getting access and running their classification algorithms, or even to obtain the test/train sets after contacting the authors. This lead to the choice of comparing our results only with standard classifiers, even though it is clear that, with the exception of the meta-profiles method, the obtained results are lower than the ones obtained in [3, 4, 8]. Note, however, that all these approaches used feature selection by first selecting a small set of genes, and this preprocessing step was not applied in the biclustering-based classifiers here proposed.

The standard classifiers tested on the original dataset outperformed most of the previous described classification methods based on biclustering (TABLE III). Nonetheless, if we compare to these results the ones obtained for the meta-profiles method (Section II.D.2), its prediction accuracy obtained with LOO CV is in the same range shown here. It is also possible to observe that the classifier based on a decision tree has the lowest prediction ability, and it is not significantly higher (p-value > 0.16) than the biclustering-based kNN based on symbol pairing with time-lags (Section II.D.1.c), which presents a prediction accuracy of 69.23% (LOO) and 68.08% (5 x 4 fold) for a maximum time-lag of 1 time point in both directions.

When applying the classifiers on a discretized version of the data, we see that the classifiers' performance decreases significantly (TABLE IV), remaining below the 60% of prediction accuracy. These evidences suggest that this kind of classifiers cannot deal well with discretized data of this type (especially with the particular characteristics discussed previously). In fact, for example, the biclustering-based kNN classifier based on symbol pairing considering time-lags (II.D.1.c) outperforms significantly all these standard classifiers (p-value < 0.05) when acting upon these discretized data.

V. CONCLUSIONS AND FUTURE WORK

Most of the results pointed to a singular characteristic of this dataset: good responders have a significant number of similar biclusters in common with other good responders, but also with the bad responders. These shared similar biclusters might include characteristic disease expression signatures, common to all RR-MS patients, a fact that shall be further investigated. Bad responders, however, show evidences of having few similar biclusters in common, beside the ones also shared with the good responders group. This fact suggests that there are different expression signatures associated to a poor response to IFN-b treatment or an absence of signature present in good responders, a probable result of differences in the fragile balance of several pathways associated to the disease and/or treatment response. This idea is a possible justification of the criteria used for the meta-profiles method, which lead to a prediction accuracy over 86% (and up to 94.23% with the most significant biclusters).

Other challenges include the class unbalance, biasing the prediction towards the good responders, and the reduced number of time points when compared to the number of genes, possibly resulting in data overfitting.

TABLE V - Summary of the main results for the developed biclustering-based classification methods.

	KNN – Bicluster Similarities	*KNN – Profile Similarities*		*KNN –Profile Similarities (Kernel)*		*KNN – Element of Bicluster*	*KNN – Symbol Pairing*	*KNN – Symbol Pairing with Time – Lags (= 1)*	*Meta - Profiles*	*Meta – Biclusters (1000; KNN)*
		Not Filtered	Filtered	Not Filtered	Filtered					
Prediction Accuracy (LOO) (%)	59.62	50.00	63.46	46.15	63.46	61.54	57.69	69.23	88.46	67.31
Prediction Accuracy (5 x 4 fold) (%)	61.92	52.69	46.54	46.54	57.31	61.54	60.77	68.08	60.77	65.39

However, a possible solution to overcome this issue lies on feature selection prior to biclustering, eliminating non-discriminative genes. Additionally, a common problem to clinical time series analyses is the reduced number of patients, also introducing important inconsistencies (especially when using a k-fold cross validation scheme, with k small, causing a loss of a significant number of training instances).

The developed biclustering-based classifiers revealed potentialities and challenges. The time-lag consideration was seen to improve significantly the prediction accuracy (p-value = 0.0025, paired t-test). Including also the possibility of different state duration (a given patient might remain in an expression state longer than the others), taking into account the patient-specific response rate in full, would probably improve the classifier's performance. The meta-biclusters classifier should be further explored, as it presents some important potentialities, not fully studied in this work. Other similarity measures between biclusters or sets of biclusters shall be also explored, to allow the (ideally direct) computation of the score between patients. We note also that, to our knowledge, biclustering has never been used before in classification of clinical expression time series, and kNN was widely shown to outperform other classifiers in classification problems involving time series data [15].

Although the precision accuracies obtained for the MS dataset are not as high as desired, we highlight that IFN-b therapy is, currently, the standard treatment for MS. Therefore, if a classifier is able to correctly predict the patients response in a percentage higher than the proportion of good responders in the population, then it presents a significant advantage. In this case, as the proportion of good responders is 61.54%, we can consider a prediction accuracy of approximately 70% as acceptable. However, we must separate two situations: the false positives (bad responders classified as good responders, thus receiving the treatment) and the false negatives (good responders missing the treatment). Given the negative side effects and the arising of alternative therapies, the classification should favor the bad responder classification. This means that we should minimize the false positive rate, thus avoiding useless and possibly harmful treatments, allowing for the patients to an earlier change to different forms of treatment for their particular situation.

Since it is possible to visualize the temporal profiles of biclusters, it would be interesting to search for differentially expressed genes between the two classes, and to map these expression profiles to specific time points. This could contribute to new insights on the response to IFN-β treatment.

Recently, the concept of triclustering associated to multiple time series analyses was proposed by [16]. Essentially, this is an extension to the presented CCC-Biclustering algorithm developed in [11], where the biclusters of genes and consecutive time points are now grouped through a third dimension: the training instances, in the case of multiple time series classification. This means that a tricluster is formed by a repeated bicluster across patients (in a clinical example), just as a bicluster is formed by a repeated profile (set of symbols on consecutive time points) across genes. In this context, each tricluster would have an associated class distribution, since this is a supervised learning task, and, with a committee of classifiers (weighted voting) the classification of new patients could be performed by comparing each of the test patient's bicluster with the ones representing each of the training triclusters. This idea is on the basis of our main future work direction.

Finally, we note that the proposed biclustering-based strategies revealed potentialities that shall be further explored in other (clinical) time series classification problems (with more instances and/or time points) and in other data mining tasks.

References

1. Hemmer, B., J.J. Archelos, and H.P. Hartung, *New concepts in the immunopathogenesis of multiple sclerosis.* Nat Rev Neurosci, 2002. **3**(4): p. 291-301.
2. Sturzebecher, S., et al., *Expression profiling identifies responder and non-responder phenotypes to interferon-beta in multiple sclerosis.* Brain, 2003. **126**(Pt 6): p. 1419-29.
3. Baranzini, S.E., et al., *Transcription-based prediction of response to IFNbeta using supervised computational methods.* PLoS Biol, 2005. **3**(1): p. e2.
4. Costa, I.G., et al., *Constrained mixture estimation for analysis and robust classification of clinical time series.* Bioinformatics, 2009. **25**(12): p. i6-14.
5. Bar-Joseph, Z., *Analyzing time series gene expression data.* Bioinformatics, 2004. **20**(16): p. 2493-503.
6. Miller, M.B. and Y.W. Tang, *Basic concepts of microarrays and potential applications in clinical microbiology.* Clin Microbiol Rev, 2009. **22**(4): p. 611-33.
7. Androulakis, I.P., E. Yang, and R.R. Almon, *Analysis of time-series gene expression data: methods, challenges, and opportunities.* Annu Rev Biomed Eng, 2007. **9**: p. 205-28.
8. Lin, T.H., N. Kaminski, and Z. Bar-Joseph, *Alignment and classification of time series gene expression in clinical studies.* Bioinformatics, 2008. **24**(13): p. i147-55.
9. Mukherjee, S. and S. Mitra, *Hidden Markov Models, grammars, and biology: a tutorial.* J Bioinform Comput Biol, 2005. **3**(2): p. 491-526.
10. Prelic, A., et al., *A systematic comparison and evaluation of biclustering methods for gene expression data.* Bioinformatics, 2006. **22**(9): p. 1122-9.
11. Madeira, S.C., et al., *Identification of regulatory modules in time series gene expression data using a linear time biclustering algorithm.* IEEE/ACM Trans Comput Biol Bioinform, 2010. **7**(1): p. 153-65.
12. Mitchell, T., ed. *Machine Learning*. 1997, McGraw-Hill.
13. Kohavi, R. and F. Provost, *Glossary of Terms.* Editorial for the Special Issue on Applications of Machine Learning and the Knowledge Discovery Process, Machine Learning, 1998. **30**(2/3): p. 271-274.
14. Kohavi, R. *A study of cross-validation and bootstrap for accuracy estimation and model selection.* in *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*. 1995. San Francisco, CA.
15. Ye, L. and E. Keogh, *Time series shapelets: a new primitive for data mining.*, in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2009: Paris, France.
16. Gonçalves, J.P., Y. Moreau, and S.C. Madeira, *Time coherent three-dimensional clustering: unraveling local transcriptional patterns in multiple gene expression time series (Abstract and poster)*, in *Ninth European Conference on Computational Biology (ECCB 2010)*. 2010.