Methods and tools for mining multivariate time series

De Gouveia da Costa Cachucho, R.E.

Chapter 5

ClaRe: Classification and Regression Tool for Multivariate Time Series

Ricardo Cachucho, Stelios Paraschiakos, Kaihua Liu, Benjamin van der Burgh, Arno Knobbe

in Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD), 2018

Abstract

As sensing and monitoring technology becomes more and more common, multiple scientific domains have to deal with big multivariate time series data. Whether one is in the field of finance, life science and health, engineering, sports or child psychology, being able to analyze and model multivariate time series has become of high importance. As a result, there is an increased interest in multivariate time series data methodologies, to which the data mining and machine learning communities respond with a vast literature on new time series methods.

However, there is a major challenge that is commonly overlooked; most of the broad audience of end users lack the knowledge on how to implement and use such methods. To bridge the gap between users and multivariate time series methods, we introduce the ClaRe dashboard. This open source web-based tool, provides to a broad audience a new intuitive data mining methodology for regression and classification tasks over time series.

5.1 Introduction

Over the past few years, there is an increased interest in the analysis of multivariate time series data. A great deal of this interest is motivated by advances in sensor technology. In many application areas, deploying sensors for continuous monitoring has become a common strategy. Over the last 10 years, sensors are becoming more accurate, with better data communication protocols, smaller and last but not least, cheaper.

As a motivating example of the great developments in sensing technology, consider the UvA BiTS sensor system [11]. This bird tracking system currently weighs around 10 grams, is powered by a solar panel, and integrates a GPS and a tri-axial accelerometer. This bio-logger, designed for birds with at least 300 grams is capable of collecting, saving and transmitting the animal movements and overall migration. This sensor system ends up weighing less than $< 3\%$ of the bird body mass. This is an example of a system that is less invasive than traditional sampling methods in ecology, which normally involve multiple captures and releases. Therefore, both the possibilities for ecology studies and the amount of available data has expanded: more species, exact migrations, flight strategies, bird activities and foraging strategies. But how to efficiently and intuitively explore sensor data, still remains a data science challenge.

From the data science perspective, sensor systems will produce time series data. In the case of sensor networks, multiple variables are collected simultaneously, producing multivariate time series. Adding to that, when collected continuously, these datasets lead to big data challenges. This raized challenges to the data mining community, on how to deal with large multivariate time series. These challenges have attracted the attention of many researcher and lead to a vast literature on time series mining [24, 14]. With the exception of a few good examples [32, 10], there is still a gap between most of these methods and the potential end users, who may lack a technical background to implement them.

Most of the sciences based on empirical observations have the potential to benefit from technological advances in sensor systems:

- children can be monitored continuously to study their social competence [99, 100];
- environmental sciences can benefit from continuous sensing [11];

- civil engineering can develop predictive maintenance of infrastructures using sensor networks [68, 98, 67];
- life sciences and health are already heavily supported by machinery that uses sensors to measure all sort of phenomena [40, 106, 95].

A common link between all the above-cited publications is that they rely on sensor monitoring systems for their continuous sampling methodologies. The continuous nature of the measurements, lead to large multivariate time series datasets. As a consequence, the traditional data analysis tools based on classical statistics are commonly not applicable to this kind of data. This leads to an opportunity to shorten the gap between the data science community and empirical sciences, if we are able to create the appropriate tools.

One could argue that the data mining community is already encouraging the publication of source code and data associated with publications. However, without a deep knowledge on the published method and the language used to implement the code, such released source code targets only a limited audience. Another very significant effort to make machine learning methods more accessible is the release of packages with collections of algorithms, such as Scikit-learn [77] for Phyton or Caret [56] for R. The downside of such packages is the need to be proficient both in the programming language that implements the package of methods and the need to know how to build a data science methodology around the chosen method. At last, there are tools for a broad audience such as Weka [32], MOA [10], Knime [8], JMulTi [55] and SPSS [38], which are intuitive and provide graphical user interfaces. The problem with such tools is that upon the development of a new method, these tools are not flexible enough to easily incorporate them. Furthermore, focusing on multivariate time series, most of them are not designed to analyze this kind of data.

Our proposal to bridge the gap between new methods and a broad audience, is to build easily accessible web-based tools, with a user interface. Such tools require no installation, are platform-independent and can be highly intuitive. Intuitive, because most people already have been exposed to hundreds of web pages and know how to read and navigate them. With an accessible GUI, these tools will broaden the potential audience to non-experts in data science. Additionally, using the web interface allows us to present such tools as Software as a Service (SaaS).

As a motivation, consider the example of a healthy aging study, developed by a team with a biomedical background [95]. The study used multiple sensor platforms in order to predict the participant's activities and energy

expenditure. This resulted in a dataset where a team of life science researcher is left to deal with multivariate time series. Therefore, it is essential to have at their disposal a tool that implements a data mining methodology that allows them to run experiments and model such multivariate time series data.

In this paper, we propose *ClaRe*[1], a *Cla*ssification and *Re*gression tool to model supervized multivariate time series. This SaaS tool adopts the *Accordion* algorithm from the previous chapter, to learn informative features and allows users to learn regression and classification models from multivariate time series with mixed sampling rates. Its intuitive web-based interface provides options of importing, pre-processing, modeling and evaluating multivariate time series data. In every step, plotting and saving data or results are allowed. Adding to the aforementioned, both source code and experimental data[2] are made openly available.

5.2 Tool Overview

ClaRe is a web-based tool that incorporates all the necessary steps for modeling time series with mixed sampling rates. Such time series are often collected from a network of sensors that measures complex phenomena. The output of such sensors are often multiple files that have variables measured at different rates and thus have special needs:

- pre-processing needs to include synchronization and merging;

[1] `http://fr.liacs.nl:7500`
[2] `https://github.com/parastelios/Accordion-Dashboard`

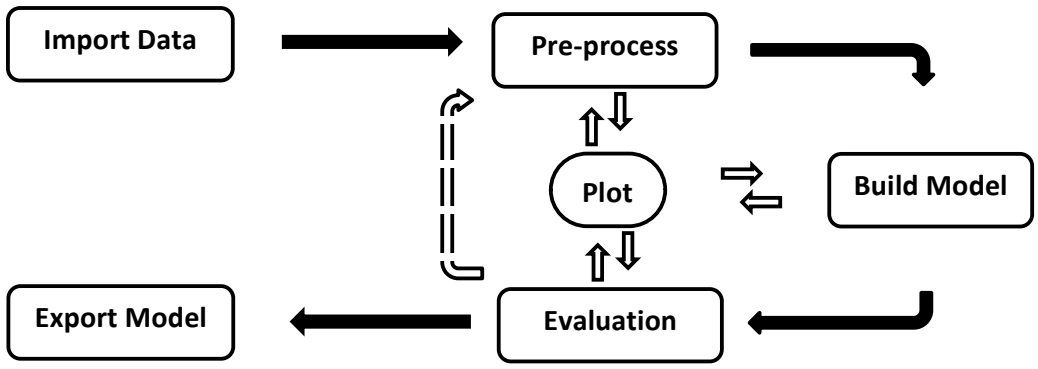

Figure 5.1: An overview of *ClaRe* tool architecture.

- plotting needs to be done using sampling techniques due to the size of such time series;
- learning strategies that take into account the temporal nature of the data;
- evaluation alternatives to cross-validation that reflect the true accuracy of the time series models.

Our tool is novel due to the capacity to deal with the challenges above, which are recurrent when dealing with sensor data.

From a technical perpective, ClaRe also presents benefits in terms of development and deployment. Both front end and server are developed with R, using the *R Shiny* package [18]. This package provides a framework to interact between client and server side through R-scripts. As a result, the tool was easy to implement since only one programming language is used to manage both server and front end. From the deployment perspective, ClaRe's main advantage is its compatibility with all modern web browsers.

ClaRe's design presents an experimental methodology as shown in Figure 5.1. One can import and pre-process time series data, build regression or classification models, evaluate them, and export the results. The user can follow the proposed methodology intuitively, using web components that adjust to the user choices and guides the user troughout the data mining methodology. Each panel will be enumerated and explained below, following the CRISP-DM standards of the data mining methodology [107].

Import: When the user accesses the tool online, they are welcomed to the tool by the *Import* panel. To start, the user can upload predictors and target in a single or separate files (see Figure 5.2). In this panel the user can get a preview of the data available and descriptive statistics for all the variables available.

Pre-processing: Having imported the data, the user will be intuitively guided to the following panel: *Pre-processing*. Here, the user can choose from multiple pre-processing tasks, both generic for all sorts of datasets and specific to sensor-based multivariate time series. The generic tasks include selecting the variable the user wants to consider as a target, normalizing datasets, removing outliers. As for tasks that are more specific to multivariate time series datasets, one can merge multiple files into one dataset, synchronize data from different sensors and manage missing values.

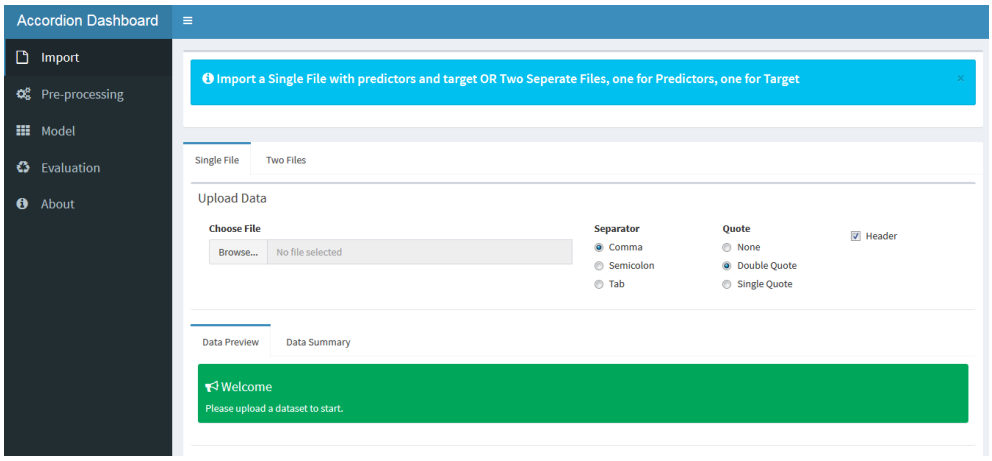

Figure 5.2: *ClaRe* dashboard user interface: Import tab.

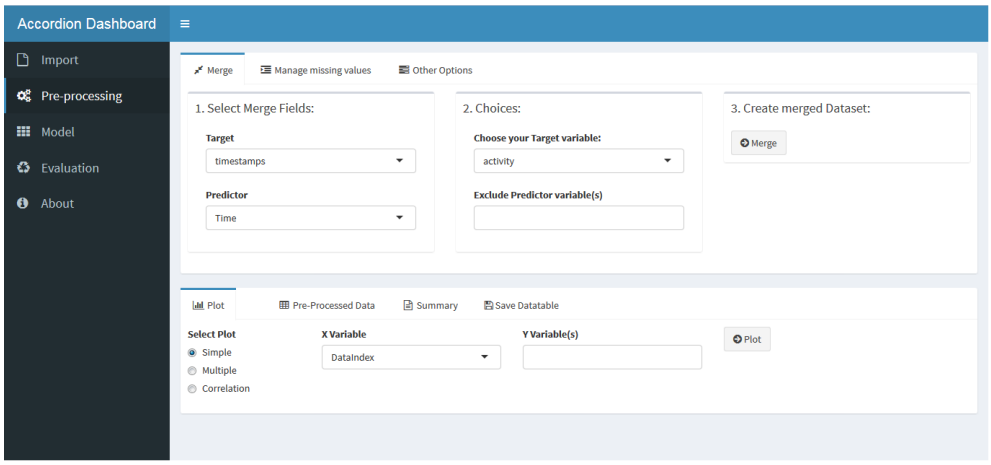

Figure 5.3: *ClaRe* dashboard user interface: Pre-processing tab.

As an example of a common pre-processing task in multivariate time series, consider a sensor network where the predictors (e.g. accelerometers) are collected and saved into one file and the target (e.g. a persons activities) into a different file. The user would want to select the target and merge both files into one synchronized dataset. For that purpose, with ClaRe the user can choose the relevant variables and *merge* them with ease (see Figure 5.3). Please note that when the user selects the target, there is a selection of which model will be used: if the target is nominal the following panel turns into a *Classification* panel (see Figure 5.5); if the target is numeric the following panel turns into a *Regression* panel (see Figure 5.7).

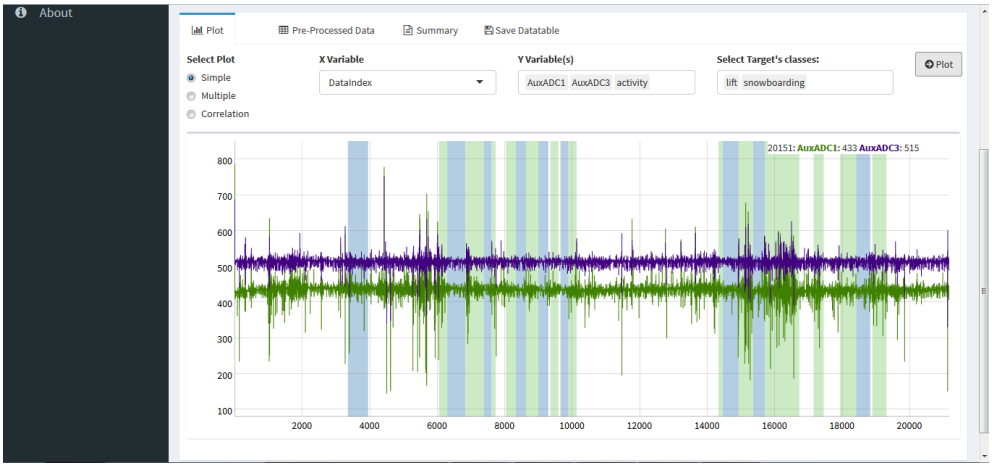

Figure 5.4: *ClaRe* dashboard user interface: Plotting tab.

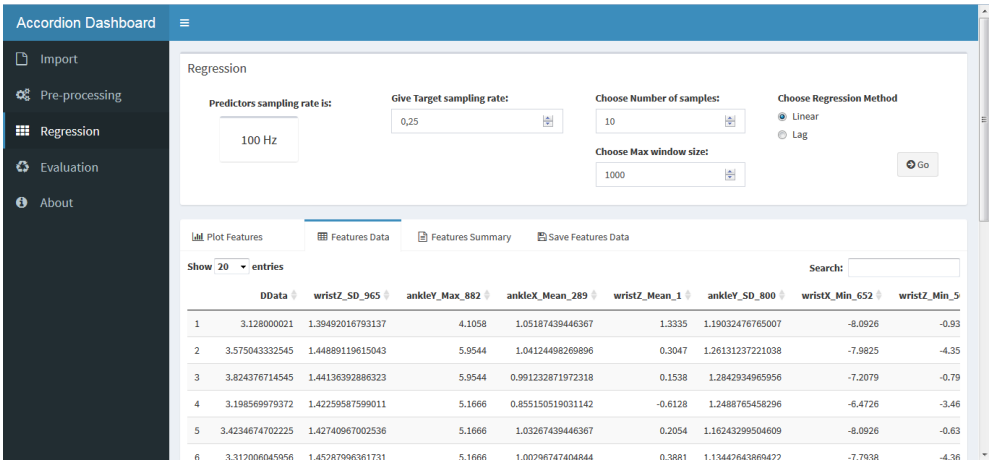

Figure 5.5: *ClaRe* dashboard user interface: Model tab.

Furthermore, there are multiple options to deal with missing values for time series, such as interpolation and repeating values. The panel of pre-processing is also allows more common tasks such as outlier removal, data normalization and conditional selection of variables.

In every step of the pre-processing, there are two useful inspection panels worth mentioning; plotting and saving. The plotting panel has multiple interactive functionalities, such as real time zoom in/out and target shading (see Figure 5.4). These plots are implemented using R's *dygraphs* package [97], which is capable of dealing with large datasets. At any time, the pre-processed dataset can be previewed and saved for further usage, allowing the experiments to continue over multiple user sessions.

Model: As mentioned before, after choosing a numeric or nominal target, this panel changes into a regression or classification setup, respectively. The available regression models are a linear regression model and a lag regression model. As for the classification task, the available model is a decision tree. Both classification and regression models construct and select aggregate features as described in the previous chapter.

Accordion can be tuned with multiple parameters, which are available in the *Regression* or *Classification* panels (see Figure 5.5). For both classification and regression, one can tune the target's sampling rate, the maximum window size and the number of samples used to perform a greedy search for aggregate features. Additionally, in regression there is an option for which regression method to use (linear or lag). To ease the users mining process, the parameter defaults are computed automatically, according to the sampling rates in the original files.

After having the parameters tuned, the users can start the learning procedure of each model by clicking a Go-button. Then, *ClaRe* starts running the Accordion algorithm on the server side of the tool and in the front end a loading-cycle is displayed until the model is constructed. After the model is built, the user can have a first inspection of the selected features by visualizing them, inspecting descriptive statistics or checking which are the variables, aggregation function and window sizes that have been selected.

Evaluation: This panel allows the users to obtain detailed cross-validated evaluations of the constructed model. An added feature is cross-validation with Leave One Participant Out (LOPO) for models using multiple participants. Such type of scenarios can be found in applications such as activity recognition for multiple people [12, 14] or birds in the examples given in the introduction. With LOPO, the model is built multiple times, leaving each time one participant out of the learning process to validate. This evaluation method is especially important to assess the real accuracy of models, when the dataset instances are not independent and identically distributed.

The evaluation of results is different for regression and classification models. For regression, it provides a full summary of the computed coefficients and model errors. For classification, it returns the confusion matrix and the associated accuracies. Furthermore, a visualization panel is available, as presented in Figure 5.7. For both regression and classification, the user can plot the dataset with the aggregated features and the predicted target. Please note that the resulting dataset with aggregate features, has the same sampling rate as the target, which can be specified as a parameter in the *Regression* and *Classification* panel.

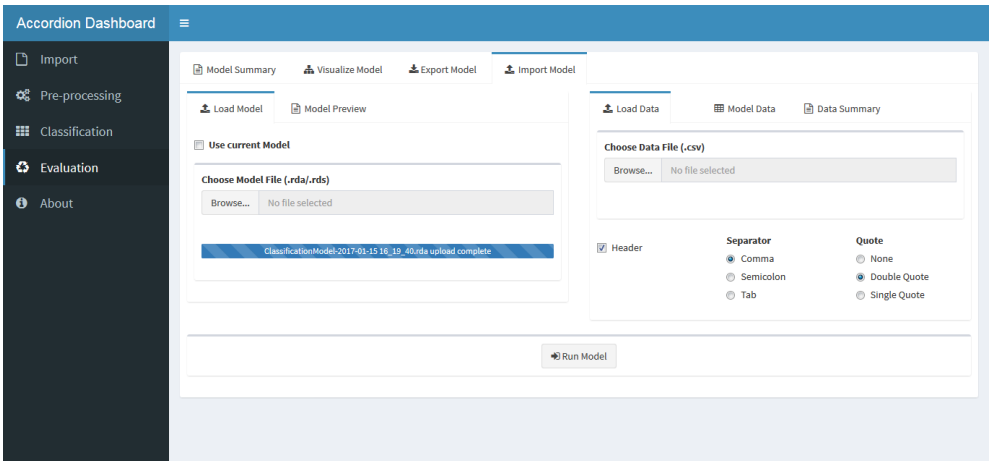

Figure 5.6: *ClaRe* dashboard user interface: Evaluation tab

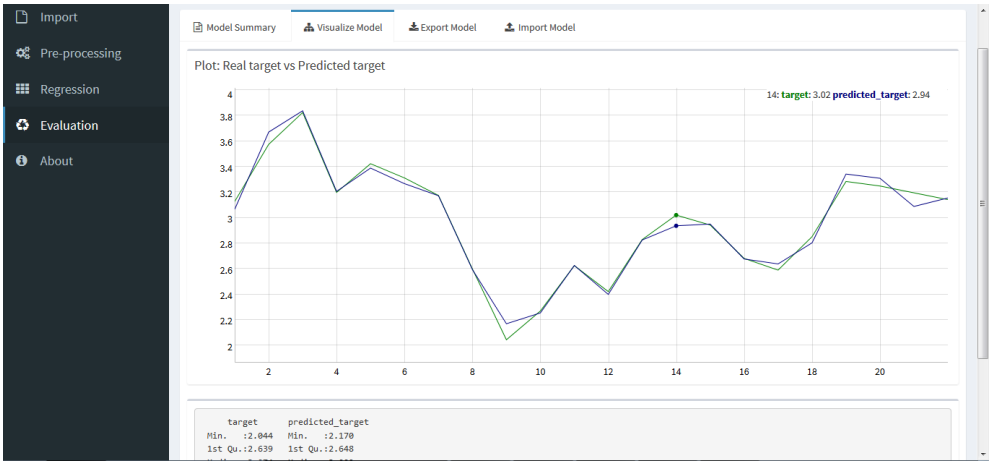

Figure 5.7: *ClaRe* dashboard user interface: Visualization of predictions

There are other functionalities that expand the evaluation possibilities. For a start, one can export the constructed model and the associated dataset. Additionally, an existing model can be evaluated using new datasets, creating a train and test evaluation scenario (see Figure 5.6). Finally, the user has the option to directly evaluate both the existing model and new datasets. All these options give the flexibility to re-visit the evaluation over multiple user sessions.

5.3 Conclusion

This paper presents an easily accessible web-tool designated as *ClaRe*. *ClaRe* is a Software as a service (SaaS), which provides any user interested in mining multivariate time series, a methodology for supervized learning. More specifically, it allows users to deal with cases when the multivariate time series data have mixed sampling rates. Making use of intuitive menus, one can easily load one of multiple files, pre-process properly sensor systems data, learn time series models and evaluate the results. Additionally, *ClaRe* is a freely distributed and open source software, that allows reproducible research in a SaaS environment.

At any stage of the mining process, interactive plotting and saving options (for models and data) are available. Being built with such options, the tool allows the mining process to be revisited over multiple user sessions, giving additional flexibility to the users. The plotting facilities of the tool are built to deal with large datasets and to give further insights at each step to the users.