SBIAGRO 2021

# Effects of agro-sensor time series approximation on plant stress detection: an experimental study

**Marcos A. de Oliveira Junior**[1,2], **Gregory Sedrez**[1], **Anderson Monteiro**[1,2], **Fernando Emilio Puntel**[1], **Gerson Geraldo H. Cavalheiro**[1]

[1]Programa de Pós-Graduação em Computação – Universidade Federal de Pelotas
Pelotas – RS – Brazil

[2]Instituto Federal de Educação, Ciência e Tecnologia Farroupilha
Santa Maria – RS – Brazil

`{marcos.oliveira,gdbsedrez,anderson.monteiro,fepuntel,`
`gerson.cavalheiro}@inf.ufpel.edu.br`

***Abstract.*** *This paper describes an experimental study on the effect of reducing time series collected from IoT electrical agro-sensors through approximation techniques, in time series classification tasks, for plant stress detection. From large sets of real data, stored in time series format, experiments were carried out to analyze: (i) performance of mathematical methods to reduce the dimensionality of time series - PAA, SAX and MCB; and (ii) Whether the application of these techniques influences the performance of time series classification models for plant stress detection, using machine learning algorithms KNN, SVM and ANN. Both in terms of data volume reduction and time series classification, the experiment showed significant improvements in terms of compression rate and accuracy, with the best result found in the use of PAA+SAX techniques for reduction and SVM for classification.*

## 1. Introduction

The usage of sensors and small boards offering processing capacity with low-energy requirements is allowing the growing of many applications in the IoT domain. The large number of devices in such applications produces lots of data that need to be stored, processed and transmitted between the application components. The time series format is often used in these applications to represent a set of data collected by a device, such as sensors, over a period of time [Blalock et al. 2018]. One characteristic of data collected by sensors is large volume and/or dimensionality. From this, algorithms emerge in order to synthesize time series of data, in order to facilitate the communication, processing and storage of these data [Krawczak and Szkatuła 2014].

Agriculture is one of the many areas that benefit from these techniques, as sensors are installed in plantations, grain cycles, and agricultural machinery [Zhang et al. 2020]. However, in most cases, sensors and stations installed in agricultural environments do not have wide availability of resources, such as storage space, or even limited hardware for data transmission. One of the important applications in the area of agriculture is related to activities that involve the application of water resources. The concern with the sustainability and finite availability of this resource implies solutions that address this issue, proposing an efficient irrigation control, since both scarcity or excess of water can cause problems for plantations. Still, in relation to the application of agricultural inputs, there is a big economic issue related, also, to an intelligent application of the products. To collect this information on water stress and obtain a better basis for decision-making, many solutions provide for the installation of sensors in different regions of the field and/or even on the plants for data collection and transmission. In many cases, data collection and transmission is performed every few seconds, generating a large amount of data. Due to the large amount of data generated, there are currently several data reduction techniques applied to agriculture, which use mathematical tools, such as time series approximation, to reduce the volume of data to be manipulated.

With a range of available techniques (reduction, compression, indexing, approximation, among others) it is extremely important to evaluate these in real applications. Because of this, this article aims to evaluate methods of approximation of time series, from real data, collected in an experiment of electrophysiology of bean plants [Toledo 2019]. For this, mathematical techniques were implemented to approximate time series known in the literature, in order to reduce the initial dataset in different ways and analyze the impact of each reduction format. Based on the concepts presented above, an experiment and a discussion of the results were developed, with the purpose of measuring and analyzing the impact of the reduction of time series, in machine learning tasks, specifically in model training and classification of time series.

The paper is organized as follows: Session 2 presents works related to the approximation of time series; Session 3 presents characteristics of the used dataset and the description of the experiment performed; Session 4 presents the discussion of the obtained results; and Session 5 presents the conclusion and future research possibilities.

## 2. Literature Review/Related Work

Time series can be generated from any measurement, in a time interval, where often the amount of data is immense. The approximation of time series is extremely important, both to reduce the volume of stored data and to reduce the amount of data transmitted and processed. In several applications, such as the Financial Market [Xin et al. 2019] and the Health area [Tobore et al. 2020], time series are widely used in solutions that generate a large volume of data in short periods of time. Because of this, data approximation techniques are valuable tools to obtain satisfactory system performance.

However, unlike the areas mentioned, where, generally, information collection, transmission and processing are carried out on computers with wide availability of resources, in agriculture it is important to use time series approximation algorithms to improve sensors scattered in crops, with scarce computational resources. Solutions with an agricultural bias, such as [Al-Qurabat and Kadhum Idrees 2020], use time series approximation algorithms for data reduction, storage and transmission, and also with low sensor energy consumption and avoiding redundant data. In these solutions for agricultural sensors, it is also observed the use of clustering algorithms when receiving the data, seeking better performance and preservation of sensors and stations in [Al-Qurabat and Idrees 2019] crops.

Currently, there are different algorithms for approximation of time series that can be used in the most diverse areas. Thus, in this work, an evaluation of some traditional approximation algorithms for time series is proposed, in combination with classification algorithms, in order to identify the effect of approximation on classification tasks. Analyzing different aspects of the behavior of the algorithms, a comparative analysis between three time series approximation strategies, combined with three classification algorithms, in an experiment carried out with real data from the electrophysiology sensing of common bean plants is presented below.

## 3. Materials and Methods

In this section, the characteristics of the data set used are presented, as well as a brief detailing about the time series approximation techniques implemented.

### 3.1. Experiment dataset

In our case study, we used the data collected in an experiment at the Plant Cognition and Electrophysiology Laboratory (LACEV), at the Federal University of Pelotas. The experiment detailed in [Toledo 2019] presents an electrophysiological analysis of bean plants, of the species *Phaseolus vulgaris L.*, cultivar BRS Expedito, through a benchtop experiment. In an environment with controlled conditions (such as humidity, temperature, and lighting), bean plants were cultivated and information on their electrical activity was collected through sensor electrodes connected to the plant stems. In order to analyze the variations in electrical activity, stimuli, such as controlled changes on soil humidity and salinity, were applied and the electromas (set of electrical activities in plants) were analyzed before and after the stimuli inductions.

The collected data were stored in time series format, with an average sampling rate of 450,000 samples every 2 hours of measurement. Measurements were taken for 2 hours before application of a stimulus and 2 hours after application. Thus, for each plant/sensor, two time series of 450,000 points were generated, one referring to the state of the plant before application and one after. For classification purposes, a value of 0 was assigned to the time series collected before the stimulus, to indicate absence of stress, while a value of 1 was assigned to the time series collected after the application of a stimulus, representing stress. For this study, measurements taken before and after the application of stimuli in 37 sensors were made available, with a total of 74 files in text format (*.txt*), containing a time series per file, of 450,000 lines, on average. Each line containing the value of a floating-point number. Each file, with these settings, has around 6 MB, with the total size of the data set available for the experiment being 441.83 MB.

### 3.2. Time Series Approximation

Given the large volume of information, for the training of machine learning models, even in the work of [Toledo 2019] it was necessary to reduce the data. In the aforementioned

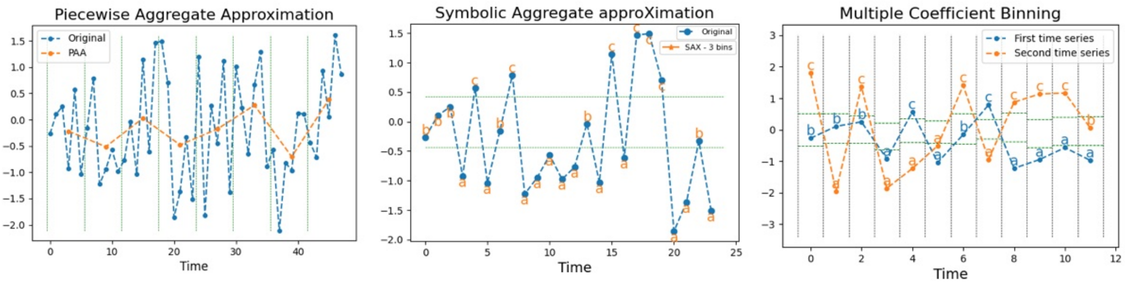

**Figure 1. Illustration of approximations with *PAA*, *SAX* and *MCB*.**

study, the authors propose to reduce the dimensionality of a data set by segmenting all time series into intervals of fixed size of 30,000 points. Each interval is represented by a tuple containing three information: the maximum and the minimum value annotated in the interval, and the average of all values. In this experiment, for performance analysis purposes, the same interval size was used.

In order to explore other possibilities for reducing the volume of data, in the present work, some alternatives for the representation of time series were investigated. The objective here is to analyze the impact of these alternatives on model training and compare them with the technique used initially, in [Toledo 2019]. Thus, three approximation algorithms were chosen and implemented, as detailed in the following.

- ***Piecewise Aggregate Approximation - PAA***: it consists in the creation of an alternative representation of a time series, from the series segmentation into intervals and in the replacement of this interval by its average value [Keogh et al. 2002]. The objective of the technique is to decrease the number of points and reduce the noise of a time series, preserving the trend. The implementation of this technique can be parameterized by configuring the size of the interval to be reduced or the number of points desired in the output. For this experiment, with an interval of 30,000 points, the time series of 450,000 points were then reduced to a time series of 15 points.
- ***Symbolic Aggregate approXimation - SAX***: uses a continuous data categorization, aiming to reduce noise and capture the trend of a time series [Lin et al. 2007]. This method independently transforms each time series (in this experiment, a sequence of floating-point numbers) into a sequence of symbols. The parameters of this technique are the number of *bins*, the strategy to determine the width of the bins and the alphabet of characters. The *bins* are the number of classes into which data will be categorized. Graphically, they correspond to the horizontal segments, as shown in the figure. The strategy can be: *uniform* (all bins of the same width); *quantile*, (all bins with the same number of points); or *normal* (bins defined according to a standard normal distribution). In this experiment, the standard parameters of the technique were used, with 4 bins, the characters *{1, 2, 3, 4}* and the distribution strategy *quantile*.
- ***Multiple Coefficient Binning - MCB***: it also uses a continuous data categorization, transforming it into strings, in order to reduce noise and capture the trend of a time series [Schäfer and Högqvist 2012]. However, this method takes into account only the values within the range to be categorized, storing each time period inde-

pendently, unlike *SAX*, which classifies the ranges according to the distribution of the entire series. This algorithm, like the previous one, also receives as input the number of *bins*, the strategy to define the width of the *bins* and the alphabet. In this experiment, the same parameters of the *SAX* technique were used.

Figure 1 graphically illustrates the application of approximation techniques: on the left, an example of the application of the PAA technique, where the original time series (blue) is reduced to a time series formed by the average value of each interval (orange); in the center, an example of the *SAX* algorithm, in which bins (horizontal lines) determine the categorization of time series values (blue) for new symbols (orange); and on the right, an example of the *MCB* technique, which determines the bins according to the values of each interval, categorizing the two presented time series (blue and orange).

Since the objectives of the above algorithms are different, the first seeks to reduce the sampling rate, while the others aim at categorizing the data, in this experiment these techniques were tested in a combined way. In order to improve the data reduction, the *SAX* and *MCB* techniques were used as a complement to *PAA*, that is, the input of these techniques was the set of time series already reduced with the previous technique.

### 3.2.1. Algorithms Implementation

The implementation of the time series approximation algorithms, detailed above, was done in the *Python* language, in the *Google Colab* tool environment. The *PYTS* [Faouzi and Janati 2020] library was used, dedicated to time series classification, which provides pre-processing and utility tools, as well as implementations of several time series classification algorithms. Data from the 74 files, detailed above, were used as input, and a single file was obtained as output, for each algorithm, with the information reduced. The techniques were applied sequentially, and the *PAA* algorithm was the first to be executed, for later the *SAX* and *MCB* techniques use the output of the first, with input for their executions.

Model training was also implemented in the same programming environment, adopting the open-source library *Scikit-learn* [Pedregosa et al. 2011] to have access to tools for predictive data analysis. The objective of the training stage was to develop models capable of identifying water stress patterns (0 without stress and 1 with stress), from the data set, in order to enable the classification of future time series (new time series). The machine learning algorithms implemented were: *K-Nearest Neighbors (KNN)*, which uses the Euclidean distance between points to classify data according to their K nearest neighbors; *Support Vector Machine (SVM)*, an algorithm that searches, through the distances between points of the same set, a linear classifier that better classifies these data; and *Artificial Neural Network (ANN)*, a mathematical model inspired by neural structure. All chosen algorithms are traditional and widely used in artificial intelligence applications. For model training and performance verification, the data sets were separated into training (80%) and test (20%) sets. For the *KNN* algorithm, the default parameters were used, with *n=3*. The *SVM* algorithm was also implemented with the default parameters. The neural network, *ANN*, was implemented with two layers, 3 and 5 nodes, with the activation function *relu*. It should be noted that, once the parameters for an algorithm were defined, they were used for training and classification in all data sets.

| Approximation | Compressed Size (KB) | Compression Ratio | Space Saving |
|---|---|---|---|
| [Toledo 2019] | 36,78 | 12013/1 | 99,991675% |
| PAA | 19,90 | 22201/1 | 99,995495% |
| PAA+SAX | 2,42 | 182201/1 | 99,999451% |
| PAA+MCB | 2,42 | 182201/1 | 99,999451% |

**Table 1. Performance of time series approximations to reduce data volume.**

## 4. Results and Discussion

The analysis on the data volume aspect is presented through the metrics *Compression Ratio* and *Space Savings* [Lelewer and Hirschberg 1987]. Both were chosen because they address the issue of data size, so important for agricultural applications, as they are usually run in environments with low availability of storage resources. As for the evaluation of the machine learning models, the metric *Accuracy* was used, based on the measurements of the hits of the classifications carried out by the models. Table 1 presents the results.

### 4.1. Data Size Reduction

The main objective, when implementing the time series reduction algorithm, is to reduce the volume of data and, consequently, the requirements involved in storing information. *Compression Ratio* is a metric that makes it possible to analyze this aspect and is obtained by dividing the original file size by the compressed file size. Complementarily, the *Space Saving* metric, indicates, in percentage, the amount of space reduced by the compression. It should be noted that the values obtained in the experiment are fixed for the data sizes and for the length of the interval used. The table presents the results found for compression ratios and saved space, in relation to the original data size, 441.83 MB, for the techniques implemented here, compared to the technique used in the [Toledo 2019] study.

Regarding the reduction in data volume, it is possible to observe that the implemented techniques achieve an excellent performance, compared to the original data volume and also to the interval mathematics technique. The *PAA* algorithm allows a great reduction in the sampling rate and, when applied in combination with the *SAX* or *MCB*, due to the categorization of the data, it achieves a very high Space Saving rate, of almost 100% as shown in the table. Although the improvements, compared to [Toledo 2019], may seem small in percentage levels, these results imply a significant reduction of space in absolute values (compressed size): more than 90%.

With such a reduction, there is a great potential for the application of these alternatives in edge devices. The level of reduction provided by the algorithms means that with the same space used to store 4 hours of measurement, it would be possible to save information uninterruptedly of approximately 10 years, using only *PAA*, and 85 years, using *PAA* in combination with *SAX* or *MCB*. This high storage gain makes the use of these techniques highly viable in agricultural applications, in scenarios of scarcity of online technological resources and/or limited hardware, such as devices like the *Arduino Uno*, a simple and cheap model, but with limited memory.

### 4.2. Time Series Classification

Decreasing the volume of data can be achieved in several ways, such as decreasing the sampling rate. However, in the scope of electrophysiology of bean plants, as detailed by

| **Approximation** | **KNN** | **SVM** | **ANN** |
|---|---|---|---|
| [Toledo 2019] | ***66,35%*** | 52,51% | 52,00% |
| PAA | 46,66% | 47,82% | 46,66% |
| PAA+SAX | 60,86% | ***95,65%*** | ***53,33%*** |
| PAA+MCB | 39,13% | 60,86% | 51,02% |

**Table 2. Performance of time series classification algorithms.**

[Toledo 2019], electrical variations can occur at very small intervals (milliseconds), so a high sampling rate is important. Therefore, a reduction strategy that does not impact data quality is necessary. Thus, the main scientific contribution brought by this work is precisely to demonstrate possibilities of data reduction, without compromising the quality of the stored data, that is, without making the use of data unfeasible.

Likewise, compared to the interval mathematics technique, the reduction techniques investigated in this work present significant improvements in terms of performance in model training. Table 2 presents the results obtained, in relation to the accuracy, regarding the performance of the implemented time series classification algorithms. The best accuracies for each algorithm are highlighted. For the *KNN* and *SVM* algorithms, the value of a single execution was considered, since both consider the distances between the points for classification, therefore, there is no variation between one execution and another. As for the training of the *ANN* algorithm, for each data set the algorithm was executed (training and testing) 10 epochs, considering the average accuracy for analysis purposes. From these results, it is observed that, while for the approximation strategies *PAA* and *PAA+MCB* present results, on average, inferior to the interval mathematics, the alternative *PAA+SAX* presented very superior results, especially when combined with the *SVM* algorithm.

## 5. Conclusion and Future Work

This article presented an experimental study on time series reduction and its impact on time series classification tasks. From real data, collected by [Toledo 2019], a first step was carried out to reduce the volume of data, where three time series approximation techniques were implemented: *Piecewise Aggregate Approximation - PAA*, *Symbolic Aggregate approXimation - SAX* and *Multiple Coefficient Binning - MCB*. The reduced time series were then used as input for the training of artificial intelligence models. For training, the algorithms *KNN, SVM* and *ANN* were used. The results of the experiment were analyzed from the *Compression Rate* and *Space Saving* metrics, for the compression techniques, and the *Accuracy*, for the classification models.

The main result of the work is the high compression rate achieved with the implemented techniques, validated together with time series classification algorithms with high accuracy rate. It has been shown that some implemented techniques outperform interval mathematics, originally used in [Toledo 2019]. The best result was identified in the compression technique that combines the *PAA+SAX* methods, which allows a space-saving of 99.999451%, used in conjunction with the *SVM* classification algorithm, which achieved an accuracy of 95%. This experiment explored simple mathematical alternatives for the reduction of time series, in order to identify possibilities for use in edge devices, with low need for computational requirements and the results found, therefore, validate the use of these techniques for agricultural solutions.

Given the experimental nature of the study, the initial results obtained are considered successful and indicate interesting alternatives to be explored. As future works, it is intended to extend the analysis to other data reduction techniques. Also, explore the use of artificial intelligence techniques in the data compression stage, as well as the use of additional data in time series classification tasks.

## Acknowledgement

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

## References

Al-Qurabat, A. K. M. and Idrees, A. K. (2019). Two level data aggregation protocol for prolonging lifetime of periodic sensor networks. *Wireless Networks*, 25(6):3623–3641.

Al-Qurabat, A. K. M. and Kadhum Idrees, A. (2020). Data gathering and aggregation with selective transmission technique to optimize the lifetime of internet of things networks. *International Journal of Communication Systems*, 33(11):e4408.

Blalock, D., Madden, S., and Guttag, J. (2018). Sprintz: Time series compression for the internet of things. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(3):1–23.

Faouzi, J. and Janati, H. (2020). pyts: A python package for time series classification. *Journal of Machine Learning Research*, 21(46):1–6.

Keogh, E., Chakrabarti, K., Pazzani, M., and Mehrotra, S. (2002). Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems*, 3.

Krawczak, M. and Szkatuła, G. (2014). An approach to dimensionality reduction in time series. *Information Sciences*, 260:15–36.

Lelewer, D. A. and Hirschberg, D. S. (1987). Data compression. *ACM Comput. Surv.*, 19(3):261–296.

Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing sax: A novel symbolic representation of time series. *Data Min. Knowl. Discov.*, 15:107–144.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Schäfer, P. and Högqvist, M. (2012). Sfa: A symbolic fourier approximation and index for similarity search in high dimensional datasets. pages 516 – 527.

Tobore, I., Kandwal, A., Li, J., Yan, Y., Omisore, O. M., Enitan, E., Sinan, L., Yuhang, L., Wang, L., and Nie, Z. (2020). Towards adequate prediction of prediabetes using spatiotemporal ecg and eeg feature analysis and weight-based multi-model approach. *Knowledge-Based Systems*, 209:106464.

Toledo, G. R. A. (2019). *Caracterização eletrofisiológica do feijão (*Phaseolus vulgaris *L.) cv. BRS-Expedito sob diferentes disponibilidades hídricas*. PhD thesis, UFPel, Pelotas.

Xin, B., Peng, W., Kwon, Y., and Liu, Y. (2019). Modeling, discretization, and hyperchaos detection of conformable derivative approach to a financial system with market confidence and ethics risk. *Advances in Difference Equations*, 2019(1):1–14.

Zhang, X., Cao, Z., and Dong, W. (2020). Overview of edge computing in the agricultural internet of things: Key technologies, applications, challenges. *IEEE Access*, 8:141748–141761.