MSAX: Multivariate symbolic aggregate approximation for time series classification*

Manuel Anacleto[1], Susana Vinga[2,3][0000−0002−1954−5487], and Alexandra M. Carvalho[1][0000−0001−6607−7711]

[1] Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa. Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
alexandra.carvalho@tecnico.ulisboa.pt
[2] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, R. Alves Redol 9, 1000-029 Lisboa, Portugal
[3] IDMEC, Instituto Superior Técnico, Universidade de Lisboa, Portugal
susanavinga@tecnico.ulisboa.pt

Abstract. Time Series (TS) analysis is a central research topic in areas such as finance, bioinformatics, and weather forecasting, where the goal is to extract knowledge through data mining techniques. Symbolic aggregate approximation (SAX) is a state-of-the-art method that performs discretization and dimensionality reduction for univariate TS, which are key steps for TS representation and analysis. In this work, we propose MSAX, an extension of this algorithm to multivariate TS that takes into account the covariance structure of the data. The method is tested in several datasets, including the Pen Digits, Character Trajectories, and twelve benchmark files. Depending on the experiment, MSAX exhibits comparable performance with state-of-the-art methods in terms of classification accuracy. Although not superior to 1-nearest neighbor (1-NN) and dynamic time warping (DTW), it has interesting characteristics for some classes, and thus enriches the set of methods to analyze multivariate TS.

Keywords: symbolic aggregate approximation · time series · classification · multivariate analysis.

1 Introduction

The vast quantity of available data nowadays is posing new challenges for knowledge discovery, namely, to extract meaningful information such as significant patterns, statistics, and regularities. Temporal data, and in particular time series (TS), are now pervasive in many fields, which fully justifies the development of new methods for their analysis. A discrete TS is a series of n real-valued

* Supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) through projects UIDB/50021/2020 (INESC-ID), UIDB/50022/2020 (LAETA, IDMEC), UIDB/50008/2020 (IT), PREDICT (PTDC/CCI-CIF/29877/2017), and MATISSE (DSAIPA/DS/0026/2019).

observations, each one being measured at a discrete time $t \in \{1, \ldots, T\}$, made sequentially and regularly trough T instances of time. In this case, the i-th TS is given by $\{\boldsymbol{x}^i[t]\}_{t\in\{1,\ldots,T\}}$, where $\boldsymbol{x}^i[t] = (x_1^i[t], \ldots, x_n^i[t])$. When $n = 1$, the TS is said to be univariate; otherwise, when $n > 1$, it is multivariate.

Data representation takes a big focus on TS analyses. An abundant wealth of data structures and algorithms for streaming discrete data were developed in recent years, especially by the text processing and bioinformatics communities. To make use of these methods, real-valued TS need symbolic discretizations. Besides this, representation methods also address the TS dimensionality problem arising from the fact that almost all TS datasets are intrinsically of high dimensionality.

In contrast to univariate TS, Multivariate TS (MTS) are characterized not only by serial correlations (auto-correlation) but also by relationships between the attributes measured at the same time point (intra-correlation). Due to considering the attributes individually, their intra-correlations might be poorly captured, as shown in [4, 5]. In [6], the necessity of different TS representations for MTS classification was discussed. It was and pointed out as desirable the development of methods that consider all attributes simultaneously, taking into account the relationships between them.

This work proposes a multivariate extension of the well-known TS representation of Symbolic Aggregate Approximation (SAX) [1]. In the SAX method, the TS is normalized to have a temporal mean of zero and a standard deviation of one. A TS normalized in this manner has a Gaussian distribution [2].

If desired, Piecewise Aggregate Approximation (PAA) [3] is then applied, reducing the TS length. This technique divides the TS into w (method parameter) segments of equal length, where each segment is replaced with its average value that is further grouped in a vector representing the TS.

Assuming that the normalized TS has a Gaussian distribution [2], it is possible to divide it into equal size areas under the Gaussian curve trough breakpoints, producing equiprobable symbols. These breakpoints may be determined by a statistical table inspection. After, the discretizing process is done by associating the TS points to a (method parameter that represents the size of the symbolic alphabet) equal area intervals beneath the Gaussian curve associated to the TS to be discretized. An illustrative example of the discretizing process is shown in Fig. 1.

Having a discretized TS, a distance measure between two TS $Q = q_1 q_2 \ldots q_T$ and $C = c_1 c_2 \ldots c_T$, in the new representation space can be defined as:

$$\text{MINDIST}\,(Q, C) = \sqrt{\frac{T}{w}} \sqrt{\sum_{i=1}^{w} dist\,(q[i], c[i])^2}. \tag{1}$$

The function $dist()$ that returns the distance between two symbols is implemented using a lookup table in which the value for entry (r, c) is obtained through the following function, where β represents the breakpoints values:

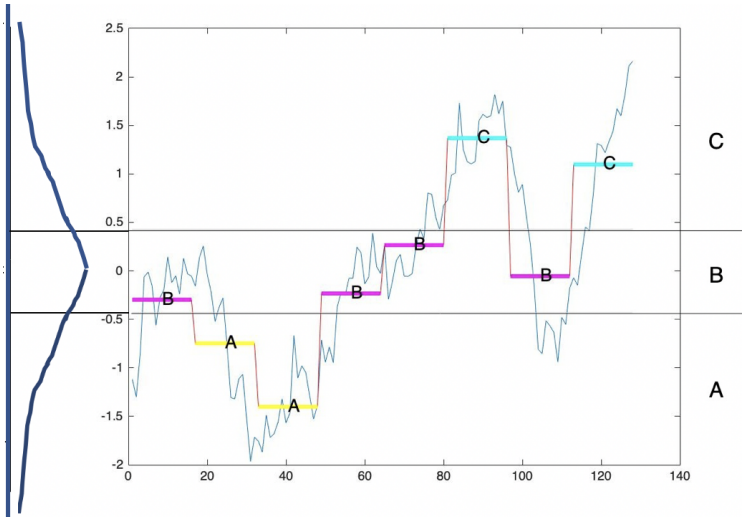

Fig. 1: The TS (in ligth blue) is discretized by first applying the PAA technique and then using predetermined breakpoints to map the PAA coefficients into the symbols. In the example above, with $T = 128$, $w = 8$ and $a = 3$, the TS is mapped to the word BAABBCBC.

$$cell_{r,c} = \begin{cases} 0, & \text{if} \quad |r - c| \leq 1 \\ \beta_{\max(r,c)-1} - \beta_{\min(r,c)}, & \text{otherwise.} \end{cases} \tag{2}$$

SAX is the only symbolic TS representation, until now, for which the distance measure in the symbolic space lower bounds the distance in the original TS space. This fact is assumed to be one of the reasons for its excellent performance [1, 2]. Nevertheless, SAX only works for univariate TS, paving the way for extending its promising results for MTS. In the literature, SAX has been applied to MTS by dealing with each variable independently, disregarding intra-correlations in the discretization process [7, 8]. We propose to explore these intra-correlations to understand the benefits of using these dependencies.

2 Materials and Methods

The method proposed in this work, MSAX, expands the SAX algorithm by first performing a multivariate normalization of the MTS. The rationale for this first step is to account for the mean and covariance structure of the data $\boldsymbol{X}[t]$, i.e., $E[\boldsymbol{X}[t]] = \mu$ and $Var[\boldsymbol{X}[t]] = \Sigma_{n \times n}$.

The normalized TS values $\boldsymbol{Z}[t]$ are given by $\boldsymbol{Z}[t] = \Sigma^{-1/2}(\boldsymbol{X}[t] - \boldsymbol{\mu})$, such that the obtained distribution has zero mean and uncorrelated variables. Assuming a Gaussian distribution, we can identify the cut points and intervals that define

equal volumes, a crucial step to identify the areas associated with symbols used in the discretization.

2.1 MSAX discretization

After the normalization step, and like in the original method, the PAA procedure is applied to each variable individually, to reduce its dimensionality. PAA can be performed individually as the resulting variables are now independent of each other, and so, intra-dependencies do not interfere with temporal ones. First, before the proper discretization of the TS values, the volumes associated with each symbol beneath the multivariate Gaussian curve are defined. With this into consideration, the following reasoning is used to define the volumes and corresponding cut-points.

Due to the normalization step, the new variables of the MTS are now uncorrelated, i.e., the covariance matrix of the TS is the identity matrix. Since the probability density function of the MTS is equal to the product of the probability density function of each variable when no correlation between the variables exist, a Gaussian distribution of $\mathcal{N}(0, 1)$ is associated to each series variable of the TS, in the same way as in the original method. Then, each Gaussian curve associated to a variable will be split using breakpoints such that the probability of each space split beneath the Gaussian curve is the same for all divisions. This procedure is done in the same way as the original method following the a parameter (that indicates the alphabet size per variable).

After the split regions under the multivariate Gaussian curve are defined trough the breakpoints intersection for each variable, this results in the variable space to be split in a grid way with each partition of the grid having the same volume under the multivariate Gaussian curve. Finally, the points of the normalized and PAA processed MTS are mapped to the multivariate split space beneath the Gaussian curve associated with multivariate TS. As a result of the entire process, a univariate discrete TS is obtained from the multivariate numerical TS.

An example of the full discretization process is given with bivariate TS normalized points X where x_1 and x_2 represent each dimension. Fig. 2 illustrates the Gaussian curve associate to this distribution. If three symbols per variable are used in the discretization process, $a = 3$, the discretization shown in Fig. 2 (right) is obtained, with a total of nine symbols. The final symbol value is obtained by the concatenation of the symbols associated with each variable (these symbols will be designated by variable symbols to distinguish from the final symbols). As an example, consider the purple partition in Fig. 2; its final symbol value is aB, directly obtained by concatenating $x_1 = a$ and $x_2 = B$ (purple partition).

2.2 Dissimilarity definition

Having introduced this new representation of MTS, a new dissimilarity measure should be defined. Two symbolic univariate TS Q and C of the same length T,

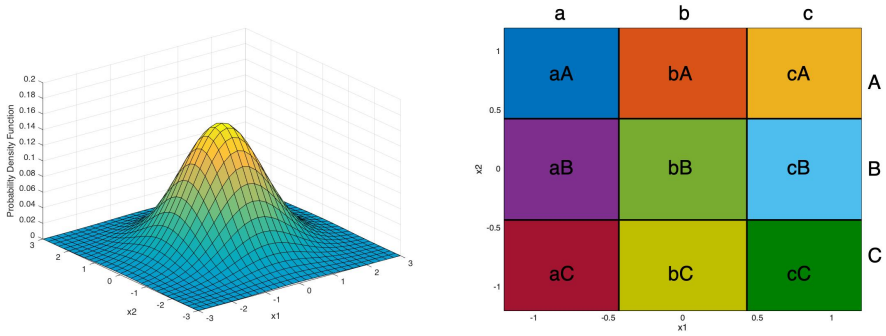

Fig. 2: On the left, plot of the probability density function or Gaussian curve with distribution $\mathcal{N}(0, I)$, for two variables x_1 and x_2. On the right, the areas associated to each symbol on the x_1, x_2 plane, for $a = 3$. Each area with a different color is associated with a symbol. For example, a point situated on the area in orange, the x_1 variable value is associated with b, and the x_2 variable value is associated with A. To this x_1, x_2 example point will be associated final symbol of bA.

obtained from an MTS with n attributes, are considered. The distance measure between two TS using the MSAX representation is given by the sum of the distances between each two-time points, for all the indexes of the TS length, where the distance between two ultimate symbols of the MSAX is obtained by the sum of the difference between the symbol of the variable associated to each variable in this representation:

$$\text{MINDIST\_MSAX}\,(Q, C) = \sqrt{\frac{T}{w}} \sqrt{\sum_{i=0}^{w} \left(\sum_{i=0}^{n} dist\,(q[i], c[i])^2 \right)}. \tag{3}$$

The distance between two symbols is calculated based on the univariate representations, and by using the corresponding distance defined originally, i.e., obtained through the same table used in the original SAX distance. This result stems from the fact that the breakpoints are the same due to the Gaussian properties.

3 Results

In this section, the MSAX algorithm is evaluated for classification tasks. Tests focused on the comparison between MSAX and the original SAX method applied to each MTS attribute separately, henceforward referred to as SAX_INDP. The behavior of both algorithms is asserted through the use of the first nearest neighbor (1-NN) classifier, varying only the input MTS representation and the respective distance measure.

Benchmark datasets for TS classification tasks included: (i) the *PenDigits* dataset, consisting in multiple labeled samples of pen trajectories; (ii) the *CharacterTrajectories* dataset, representing instead trajectories of characters from the English alphabet; and (iii) 12 datasets with different characteristics from a wide range of areas [10, 11].

Firstly, we addressed the comparison between MSAX and SAX_INDP in the PenDigits and CharacterTrajectories datasets varying both the alphabet size and the TS length reduction ratio. Results are depicted in Fig. 3. For both datasets, the accuracy of the SAX_INDP is superior to the MSAX for all parameters configurations on both plots. While fixing the w parameter, both methods on both datasets show a similar behavior by increasing the accuracy as long as the alphabet size increases. When the alphabet size is fixed, and the TS length reduction varies, the behavior differs. In the PenDigits dataset, the accuracy increases as long as the TS length reduction diminishes, whereas, in the CharacterTrajectories dataset, the accuracy remains the same as long the TS length reduction diminishes.

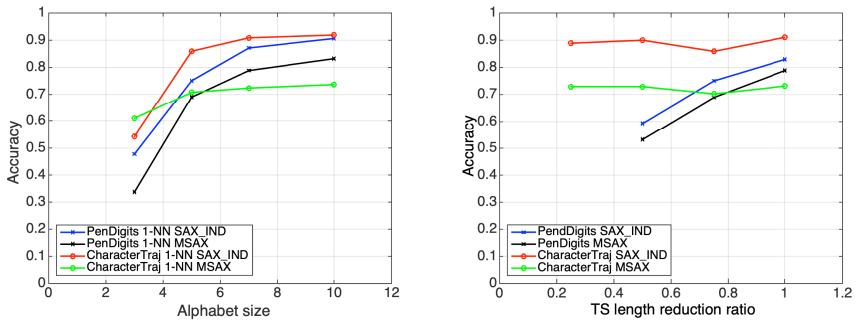

Fig. 3: Comparison between MSAX and SAX_INDP with 1-NN in PenDigits and CharacterTrajectories datasets. On the first plot, the accuracy of the methods is plotted against the parameter a; for these experiments, a fixed value of w was used. On the second, the accuracy is plotted against the TS length reduction ratio (obtained trough the parameter w); for these experiments, a fixed value of a was used.

Additional results comparing both methods were performed by testing 14 datasets with a combination of configurations from an alphabet size varying from 5 to 20 and a TS length reduction ratio from 1/4 to 1. Besides the SAX-based methods, two state-of-the-art classifiers were used: the 1-NN with Euclidean distance and 1-NN with Dynamic Time Warping (DTW). Fig. 4 presents the result corresponding to the configuration of parameters that achieved the best accuracy.

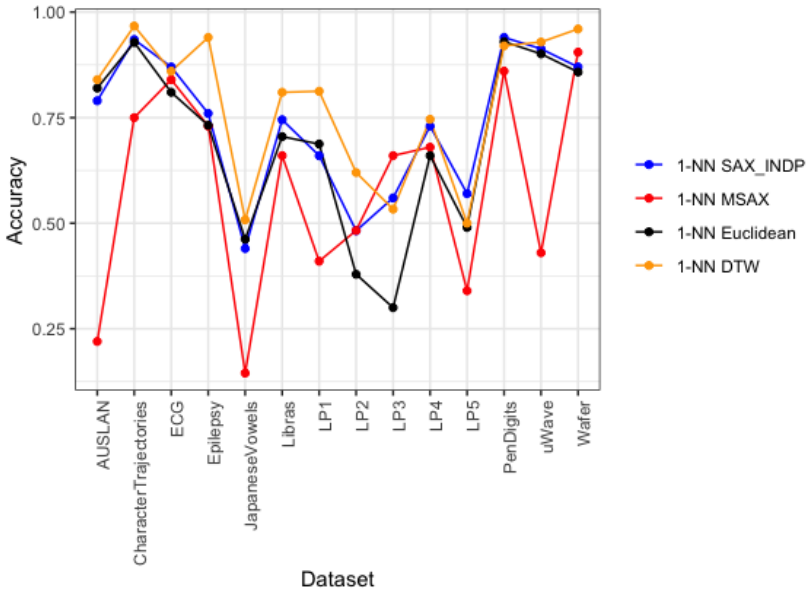

Fig. 4: Accuracy of the four classifiers: 1-NN with SAX_INDP, 1-NN with MSAX, 1-NN with Euclidean distance, and 1-NN with DTW, for 14 benchmarks datasets in TS classification tasks.

On the 14 datasets the accuracy of the SAX_INDP is superior in 12 we compared to the MSAX. In this 12 datasets, the difference is very significant in 6 of them, while in the other 6 the accuracy of both methods is very close. Regarding the comparison of the SAX-based methods with the other two state-of-the-art classifiers, SAX_INDP proves to be very competitive with the Euclidean distance, presenting a small superiority; the results are very similar in 10 datasets, whereas in 4 datasets the SAX_INDP achieves a significantly better result. Concerning the DTW distance, it surpasses, in general, the other algorithms, being the most accurate on most of the datasets. Nonetheless, SAX_INDP achieves very similar and competitive results on a significant number of datasets.

4 Conclusion

In this work, an extension of SAX for multivariate TS, named MSAX, was proposed. Its behavior was assessed in classifications tasks, comparing it with the SAX_INDP and two other state-of-the-art classifiers: 1-NN with the Euclidean distance and 1-NN with DTW. We concluded that the proposed method is overall not competitive with the SAX_INPD, the original SAX algorithm applied independently to each attribute in the MTS. Nonetheless, the obtained results have utility as benchmark values for SAX-based methods in multivariate classifications tasks. It is also noteworthy that for some datasets and specific cases,

MSAX surpasses the other techniques. As a future direction, MSAX could be evaluated more deeply in different data mining tasks, such as clustering or forecasting, in which it could be useful and achieve comparable performance with state-of-the-art methods. Possible future applications in bioinformatics include the analysis of patients' data, such as transcriptomics and also time series from electronic health records.

References

1. Lin, Jessica and Keogh, Eamonn and Lonardi, Stefano and Chiu, Bill. "A symbolic representation of time series, with implications for streaming algorithms". *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pp. 2–11, 2003.
2. Lin, Jessica and Keogh, Eamonn and Wei, Li and Lonardi, Stefano. "Experiencing SAX: a novel symbolic representation of time series". *Data Mining and knowledge discovery*, vol.15, no.2, pp. 107–144, 2007.
3. Keogh, Eamonn and Chakrabarti, Kaushik and Pazzani, Michael and Mehrotra, Sharad. "Dimensionality reduction for fast similarity search in large time series databases". *Knowledge and information Systems*, vol.3, no.3, pp. 263–286, 2001.
4. Xiaoqing Weng and Junyi Shen. "Classification of multivariate time series using locality preserving projections". *Knowledge-Based Systems*, vol.21, no.7, pp. 581–587, 2008.
5. Bankó, Zoltán and Abonyi, János. "Correlation based dynamic time warping of multivariate time series". *Expert Systems with Applications*, vol.39, no.17, pp. 12814–12823, 2012.
6. Kadous, Mohammed Waleed and Sammut, Claude. "Classification of Multivariate Time Series and Structured Data Using Constructive Induction". *Machine Learning*, vol.58, no.2, pp. 179–216, 2005.
7. Esmael, Bilal and Arnaout, Arghad and Fruhwirth, Rudolf and Thonhauser, Gerhard. "Multivariate Time Series Classification by Combining Trend-Based and Value-Based Approximations". In: Murgante B. et al. (eds) Computational Science and Its Applications – ICCSA 2012. Lecture Notes in Computer Science, vol 7336. Springer, Berlin, Heidelberg.
8. Wang, Zhiguang and Song, Wei and Liu, Lu and Zhang, Fan and Xue, Junxiao and Ye, Yangdong and Fan, Ming and Xu, Mingliang. "Representation Learning with Deconvolution for Multivariate Time Series Classification and Visualization". *CoRR*, vol. abs/1610.07258, 2016.
9. Alimoglu, F. and Alpaydin E., "Combining multiple representations and classifiers for pen-based handwritten digit recognition", Proceedings of the Fourth International Conference on Document Analysis and Recognition, pp. 637–640 vol.2, Germany, 1997.
10. Bagnall, Anthony J. and Dau, Hoang Anh and Lines, Jason and Flynn, Michael and Large, James and Bostrom, Aaron and Southam, Paul and Keogh, Eamonn J. "The UEA multivariate time series classification archive, 2018". *CoRR*, vol. abs/1811.00075, 2018.
11. Bayodan, Mustafa. http://www.mustafabaydogan.com – Mustafa Baydogan website. Last accessed 10 Mar 2020.