Time series classification using F-transform

Przemyslaw Grzegorzewski[1,2,*,†], Antoni Kędzierski[1,†]

[1]Faculty of Mathematics and Information Science, Warsaw University of Technology, Koszykowa 75, 00-662 Warsaw, Poland

[2]Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447 Warsaw, Poland

Abstract

In this paper, we propose a new methodology for time series classification. It employs two techniques: the fuzzy transform (F-transform) and the well-known decision tree classifier. A combination of these two tools appears to result in a new classification method that shows good statistical properties and could be a noteworthy alternative for considered as best for time series 1NN classifier.

Keywords

Fuzzy transform, classification, decision tree, distances, time series

1. Introduction

Time series data form an increasing proportion of the world's data supply. The omnipresence of time series and the exponentially growing size of databases resulted in an explosion of interest in Data Mining methods adapted to the specificity of time series. In typical time series mining tasks, such as indexing, grouping, classification, forecasting, segmentation, summarization, and anomaly detection, the analysis of the similarity between the series plays an important role (see, e.g. [1]). The appropriate selection of such a similarity measure may be of significant importance for the quality and effectiveness of statistical inference [2, 3].

The so-called fuzzy transform (or F-transform, for short), introduced by Perfilieva [4], is a special technique that can be used to obtain a simple approximate representation of functions that captures their essential features. The theory of F-transform was developed extensively in recent years and brought many successful applications in image processing, data analysis, and signal processing. It also seems to have interesting potential in other fields, like ordinary and partial differential equations with fuzzy initial conditions. Thus, it should come as no surprise that the F-transform found an application in time series analysis, especially for time series forecasting (see, e.g., [5, 6, 7, 8]).

The main goal of this contribution is to compare the best distance measures in the most popular 1NN method with the new classification method – random forest based on F-transform. We want to test the usefulness of the F-transforms in time series classification.

OLUD 2022: First Workshop on Online Learning from Uncertain Data Streams, July 18, 2022, Padua, Italy.

[*] Corresponding author.

[†] These authors contributed equally.

Przemyslaw.Grzegorzewski@pw.edu.pl (P. Grzegorzewski); antoni.kedzierski.stud@pw.edu.pl (A. Kędzierski)
http://pages.mini.pw.edu.pl/~grzegorzewskip/www/ (P. Grzegorzewski)
0000-0002-5191-4123 (P. Grzegorzewski)

The paper is organized as follows: in Section 2 we recall basic information on the F-transform. Next, in Section 3, we make a short introduction to time series classification and propose how to apply the F-transform for this task. Then, in Section 4, we describe the investigation conducted to examine the proposed classification method and discuss some of the experimental results.

2. Fuzzy transform

In this section, we provide only the definition and basic concepts related to the F-transform. For more details we refer, e.g., to [4].

Let us consider a continuous real function $f : [a, b] \to \mathbb{R}$. The F-transform is defined concerning the so-called *fuzzy partition* of the domain $[a, b]$ by a finite number of fuzzy sets satisfying some axioms specified in the following definition.

Definition 1. *Let $c_1 < \ldots < c_n$ denote fixed nodes within $[a, b]$, such that $c_1 = a$ and $c_n = b$ and $n \geqslant 3$. We say that fuzzy sets $A_1, \ldots, A_n$ form a* ***fuzzy partition*** *of $[a, b]$ if they satisfy the following conditions for $i = 1, \ldots, n$:*

1. $A_i(c_i) = 1$;
2. $A_i(x) = 0$ *for $x \notin (c_{i-1}, c_{i+1})$, where for uniformity of notation, we put $c_0 = c_1 = a$ and $c_{n+1} = c_n = b$;*
3. A_i *is continuous;*
4. A_i *is strictly increasing in $[c_{i-1}, c_i]$ and strictly decreasing in $[c_i, c_{i+1}]$;*
5. $\sum_{i=1}^{n} A_i(x) = 1$ *for each $x \in [a, b]$.*

The last axiom is known as *orthogonality* or the Ruspini condition. The membership functions of fuzzy sets $A_1, \ldots, A_n$ forming a fuzzy partition are called *basic functions*. It is worth noting that the shapes of basic functions are not predetermined and can be selected to meet some additional specific properties. A fuzzy partition is called *uniform* if the nodes are equidistant (i.e. $c_i = c_{i-1} + h$ for $i = 2, \ldots, n$ and some fixed h) and fuzzy sets $A_2, \ldots, A_{n-1}$ are shifted copies of symmetrized A_1 (or A_n, for details see [4]). Once a fuzzy partition is selected we can define the F-transform.

Definition 2. *Let $A_1, \ldots, A_n$ denote a fuzzy partition of $[a, b]$ and let $f : [a, b] \to \mathbb{R}$ be a continuous function. The n-tuple $(F_1, \ldots, F_n)$ of real numbers given by*

$$F_i = \frac{\int_a^b f(x) A_i(x) dx}{\int_a^b A_i(x) dx}, \quad i = 1, \ldots, n, \tag{1}$$

is a (direct) ***fuzzy transform*** *(F-transform) of f with respect to the given fuzzy partition.*

In practical applications, f is usually not given analytically. Instead, we are provided with some data points obtained from observations or measurements. Thus Def. 2 can be modified by replacing integrals in (1) by finite sums.

To be more strict, let $A_1, \ldots, A_n$ denote a fuzzy partition of $[a, b]$ and let the function $f : [a, b] \to \mathbb{R}$ be given at fixed points $p_1, \ldots, p_T \in [a, b]$, where $T > n$. We say that the set of points $\{p_1, \ldots, p_T\}$ is *sufficiently dense with respect to the fuzzy partition* $A_1, \ldots, A_n$ if for every $i \in \{1, \ldots, n\}$ there exist $t \in \{1, \ldots, T\}$ such that $A_i(p_t) > 0$. Now we are able to define the so-called discrete F-transform.

Definition 3. *Let $A_1, \ldots, A_n$ denote a fuzzy partition of $[a, b]$ and let $f : [a, b] \to \mathbb{R}$ be a function known at points $p_1, \ldots, p_T \in [a, b]$. Moreover, let us assume that the set $\{p_1, \ldots, p_T\}$ is sufficiently dense with respect to the fuzzy partition $A_1, \ldots, A_n$. Then the n-tuple $(F_1, \ldots, F_n)$ of real numbers given by*

$$F_i = \frac{\sum_{t=1}^{T} f(p_t) A_i(p_t)}{\sum_{t=1}^{T} A_i(p_t)}, \quad i = 1, \ldots, n, \tag{2}$$

is a (direct) ***discrete fuzzy transform*** *(discrete F-transform) of f with respect to the given fuzzy partition.*

In applications we usually refer to the F-transform without specifying explicitly whether it is for a continuous or discrete problem, i.e. according to Def. 2 or Def. 3, assuming that the reader can recognize which is the actual underlying concept from the context.

The F-transform (as well as the discrete F-transform) of f will be denoted by $F_n[f] = (F_1, \ldots, F_n)$, where the number F_i is called the i-th *component* of the F-transform. One can realize that the components of the F-transform are just weighted mean values of the original function f, where the weights are determined by the basic functions $A_1, \ldots, A_n$.

The F-transform itself would not be interesting enough without its inverse formula which allows reconstructing f from $F_n[f]$.

Definition 4. *Let $F_n[f] = (F_1, \ldots, F_n)$ be the direct F-transform of f with respect to a fuzzy partition $A_1, \ldots, A_n$ of $[a, b]$. Then the function $f_{F,n} : [a, b] \to \mathbb{R}$ given by*

$$f_{F,n}(x) = \sum_{i=1}^{n} F_i \cdot A_i(x), \quad x \in [a, b], \tag{3}$$

is called the ***inverse F-transform*** *of f.*

It is seen that the inverse F-transform is a continuous function on $[a, b]$. However, what's more, it can be shown that the sequence of the inverse F-transform $\{f_{F,n}\}_{n=3}^{\infty}$ converges uniformly to the initial function f as $n \to \infty$. Moreover, this result is valid both when a fuzzy partition is uniform [4] or non-uniform [9].

Thus, to conclude, while the direct F-transform $F_n[f]$ may serve as a discrete approximate representation of a function $f : [a, b] \to \mathbb{R}$, the inverse F-transform $f_{F,n}$ is a suitable continuous approximation of f.

Both of these statements make the F-transform an extremely useful tool in various fields and applications. One of them is time series analysis. Until now, it was used there mainly and successfully for prediction (see, e.g., [5, 6, 7, 8]). Our goal is to test its suitability for other tasks related to time series analysis, in particular, the classification of time series.

3. Time series classification

Time series classification is an important problem in data analysis. Although many algorithms have been proposed the nearest neighbor (NN) classifier still seems to be the most appreciated one. It is so mostly because of its simplicity and noticeably good performance in many situations. The 1NN classifiers are mainly used for time series due to the dimensionality of the data. Their high performance, especially with dynamic time warping (DTW) and its modified versions used as the distance measure has been confirmed by many experiments (see, e.g., [2, 3]).

In our study, we set ourselves the goal to make use of renowned classification methods, such as the random forest classifier, but in the field of time series. The reason we use the F-transform is to reduce the high dimensionality of time-series data. Indeed, using the F-transform we can compress time series to significantly smaller vectors and then perform the classification task on them.

Let $X = \{x_t : t \in T_x\}$ be a given time series. Obviously, X might be viewed as a function $x(t) = x_t$ defined on a fixed time interval which is not given analytically but instead some measurements x_t at points $t \in T_x$ are available. Assuming $\{x_t : t \in T_x\}$ is sufficiently dense with respect to the fuzzy partition $A_1, \ldots, A_n$ and following (2) the F-transform of X is given by $F_n[X] = (F_1, \ldots, F_n)$, where

$$F_i = \frac{\sum\limits_{t \in T_x} x_t A_i(t)}{\sum\limits_{t \in T_x} A_i(t)}, \quad i = 1, \ldots, n. \tag{4}$$

Looking at the time series representation given by (4) it becomes clear that the use of the F-transform makes it possible to reduce significantly the dimensionality of the data.

Further on we will conduct tests to check whether our approach combining decision trees and the F-transform can compete with other widespread classification tools.

4. Experimental results

To investigate how the proposed F-transform-based random forest classifier behaves and to compare it with classifiers utilizing various distances and similarity measures we conducted an extensive experimental study. It was performed on benchmark datasets from the UCR Time Series Classification Archive [10, 11]. The UCR time series repository contains 128 datasets originating from different domains and such sources as electrocardiograms, power measurements, sensor readings, spectroscopy, traffic data, simulated data, etc. Within the data, one can find cases with a highly diversified number of classes, with various time series per dataset and time series of different lengths.

In our experiment, we analyzed 29 datasets included in the UCR repository (i.e. PowerCons, Coffee, BME, SmoothSubspace, Wafer, Plane, Strawberry, ItalyPowerDemand, Meat, GunPoint, CBF, UMD, BeetleFly, Symbols, MoteStrain, ECG200, Trace, SwedishLeaf, FaceFour, Yoga, Beef, Wine, Fish, Fungi, ShapesAll, Ham, FiftyWords, ElectricDevices, BirdChicken). They have been specially selected to deal with a wide range of possible problems that can be encountered when analyzing time series, since they differ in the number of observations, the sizes of the training

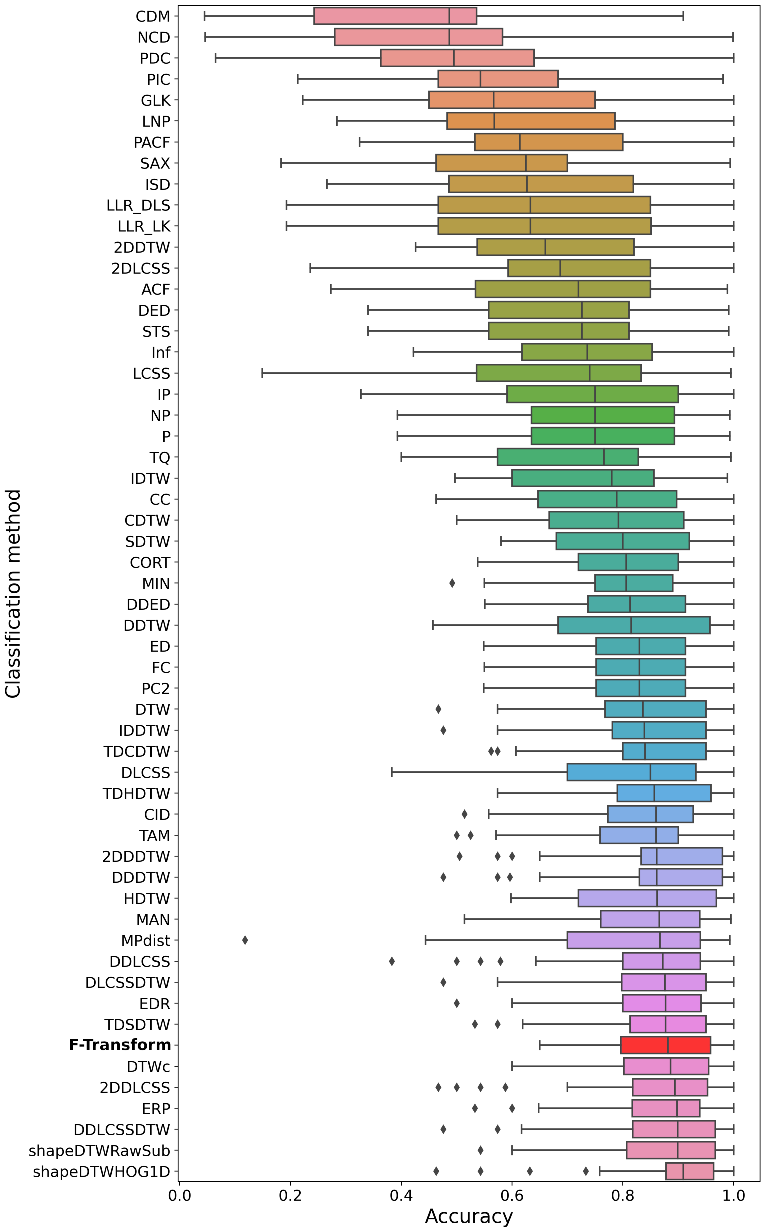

Figure 1: A comparison of the accuracy for all tested methods.

and test sets, the shape of their trajectories, etc. For instance, *PowerCons* contains data on the energy consumption of French households over two seasons - heating and summer. The collected trajectories contain clear observations of outliers related to the increased energy consumption. *Coffee* dataset contains spectrograms of two different types of coffee - Robusta and Arabica. The *BME* set is artificially generated data that represents three types of trajectories: containing a local maximum at the beginning of the series, containing a local maximum at the end of the series, and having no shots. *Wafer* contains trajectories corresponding to the records from several sensors monitoring the production process, the so-called silicon wafers, where the observation labels say whether the record describes a normal or a disturbing process. The *Plane* set contains the outlines of seven different airplane models converted into a one-dimensional time series. The thing that characterizes the contour data is often the high variability over a small period, which corresponds to small indentations in the shape. *Strawberry* contains spectrograms of fruit mousses made from strawberries or from strawberries with the addition of other fruits. For a more detailed description of the datasets, we refer to [10, 11] and the bibliography cited there.

The suggested F-transform-based random forest classifier was compared with the most effective variants of the 1NN algorithm equipped with 55 metrics and similarity measures described in [2, 3]. These distances/similarity measures could be grouped into four categories: shape-based measures, edit-based measures, feature-based measures and structure-based measures [12].

Figure 1 presents a comparison of the considered classification methods. Box-plots given there shows the distribution of their accuracy obtained for all 29 benchmarks. As to be expected, there is no definite winner but the top efficient classifiers are based on distances related to DTW and its modifications. This result is in line with the conclusions of the research conducted by Górecki and Piasecki [2, 3]. However, it should be underlined that the method based on the F-transform appears among the winning approaches. And although it ranks eighth in

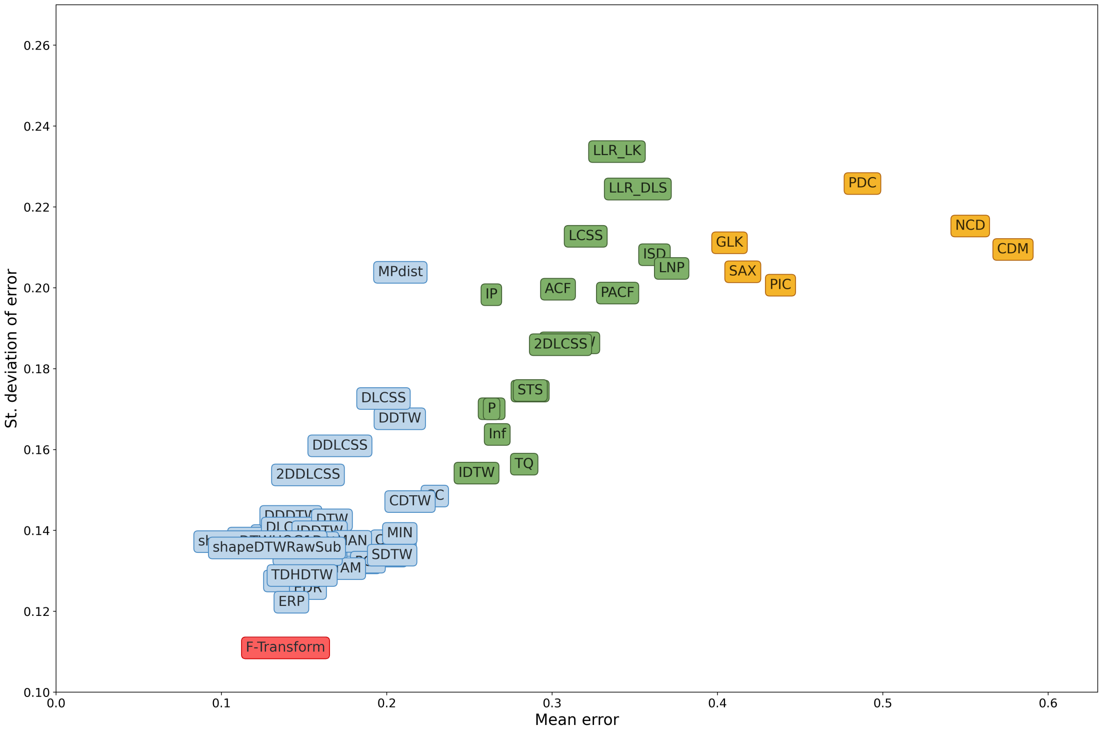

Figure 2: Mean errors and standard deviations of error for each method.

terms of the median, it is worth noting that the classifier utilizing the F-transform is the most stable method on the list. Indeed, it did not work badly on any of the 29 analyzed datasets (no outliers on its boxplot) and at the same time, it has a quite small dispersion (both range and the interquartile range).

Good properties of the method utilizing the F-transform are confirmed by the drawing in Figure 2 where each considered method is placed on a map describing two features: the mean error and standard deviation (to be more strict, the position of the method corresponds to the upper left corner of the rectangle representing the given method). As it is easily seen the proposed classification method based on the F-transform reveals the smallest standard deviation of the error and one of the smallest mean errors.

Additionally, all considered methods were grouped according to clusters, where the mean errors and standard deviations were used as features. Figure 2 shows the results of k-means clustering, where the optimal number of clusters turned out to be equal to 3. Hence we have obtained 3 disjoint groups: weak and unstable (yellow), moderate (green), and highly efficient and stable (blue). Our classification method based on the F-transform belongs to the "blue" cluster but we marked it in red to make it easier to identify in the drawing.

5. Conclusions

Our research confirmed that none of the classification methods neither any of the distance measures nor our F-transform classifier is the best for all available datasets. However, there is a group of methods performing significantly better than the others. This group includes the proposed classifier based on the F-transform. The limited space does not allow the presentation and discussion of all the results obtained, but - as we believe - the reader may be convinced of the high quality of the adsorbed method based on what is presented in this contribution. Anyway, classifiers obtained with the F-transform challenge the methods considered to be the best at present, i.e. the 1NN algorithm with an appropriate metric.

Moreover, the F-transform enables use in a classification other than distance-based methods (e.g. decision trees, logistic regression) as it significantly reduces the dimensionality of the data. Due to the F-transform, we can jump to a completely new level of classification and use completely different methods, which - as it turns out - work well. We hope that our study will also contribute to increasing the interest in the F-transform, showing its usefulness not only in forecasting (as demonstrated earlier), but also in other time series analysis tasks.

Many questions and problems are still open. In particular, we want to examine if there is a significant relationship between the fuzzy partition selection and the resulting classification. We have observed that the F-transform works effectively for the classification of spectrographic data. Hence, we want to indicate a class of time series for which the F-transform method reveals the best properties as a classifier. Finally, in the nearest future, we plan to examine how to apply the F-transform in other time series analysis problems such as cluster analysis or anomaly detection.

References

[1] C. Ratanamahatana, J. Lin, D. Gunopulos, E. Keogh, M. Vlachos, G. Das, Mining time series data, in: O. Maimon, L. Rokach (Eds.), Data Mining and Knowledge Discovery Handbook, Springer, 2010, pp. 1049–1077. doi:10.1007/978-0-387-09823-4_56.

[2] T. Górecki, P. Piasecki, An experimental evaluation of time series classification using various distance measures, Archives of Data Science, Series A 5 (2018) 1–12. doi:10.5445/KSP/1000087327/07.

[3] T. Górecki, P. Piasecki, A comprehensive comparison of distance measures for time series classification, in: A. Steland, E. Rafajłowicz, K. Szajowski (Eds.), Stochastic Models, Statistics and Their Applications, volume 294 of *Springer Proceedings in Mathematics & Statistics*, Springer Nature, 2019, pp. 409–428. URL: https://doi.org/10.1007/978-3-030-28665-1_31.

[4] I. Perfilieva, Fuzzy transforms: theory and applications, Fuzzy Sets and Systems 157 (2006) 993–1023. doi:10.1016/j.fss.2005.11.012.

[5] M. Stepnicka, A. Dvorak, V. Pavliska, L. Vavrickova, A linguistic approach to time series modeling with the help of f-transform, Fuzzy Sets and Systems 180 (2011) 164–184. doi:10.1016/j.fss.2011.02.017.

[6] M. Stepnicka, P. Cortez, J. P. Donate, L. Stepnickova, Forecasting seasonal time series with computational intelligence: On recent methods and the potential of their combinations, Expert Systems with Applications 40 (2013) 1981–1992. URL: http://dx.doi.org/10.1016/j.eswa.2012.10.001.

[7] S. Mirshahi, V. Novak, A fuzzy method for evaluating similar behavior between assets, Soft Computing 25 (2021) 7813—-7823. URL: https://doi.org/10.1007/s00500-021-05639-y.

[8] V. Novak, S. Mirshahi, On the similarity and dependence of time series, Mathematics 9 (2021) 550. URL: https://doi.org/10.3390/math9050550.

[9] M. Stepnicka, O. Polakovic, A neural network approach to the fuzzy transform, Fuzzy Sets and Systems 160 (2009) 1037–1047. doi:doi:10.1016/j.fss.2008.11.029.

[10] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, Hexagon-ML, The ucr time series classification archive, 2018. https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

[11] H. A. Dau, A. Bagnall, K. Kamgar, C.-C. M. Yeh, S. Gharghabi, C. A. Ratanamahatana, E. Keogh, The ucr time series archive, 2019. https://doi.org/10.48550/arXiv.1810.07758.

[12] P. Esling, C. Agon, Time-series data mining, ACM Computing Surveys (CSUR) 45 (2012). doi:10.1145/2379776.2379788.