On the need of unfolding preprocessing for time series clustering

Geoffroy Simon, John A. Lee, Michel Verleysen
Machine Learning Group, UCL - DICE, Place du Levant, 3
B-1348 Louvain-la-Neuve BELGIUM
{simon, lee, verleysen}@dice.ucl.ac.be

Abstract - *Clustering methods are commonly used on time series, either as a preprocessing for other methods or for themselves. This paper illustrates the problem of clustering applied on regressor vectors obtained from row time series. It is thus shown why time series clustering may sometimes seem meaningless. A preprocessing is proposed to unfold time series and allow a meaningful clustering of regressors. Graphical and experimental results show the usefulness of the unfolding preprocessing.*

Key words - Clustering, Time Series, SOM

1 Introduction

Time series have been broadly studied in many domains. During the last decades, modeling tools and analysis methodologies have raised in system identification, statistics, econometrics, data mining, machine learning and neural networks communities, to cite only a few of them. Clustering algorithms are commonly used in time series analysis, either as a preprocessing to another model or as a method of feature or rule extraction, for example.
Clustering of time series usually copes with vectorial representation of a series; the so-called *regressors* are obtained by creating blocks of successive values in a sliding window approach. Recently, it has been claimed that the clustering of time series regressors is meaningless [1], because the specific information contained in the series is lost in the clustering process. The meaningfulness of applying clustering methods on time series regressors is also questioned in [2, 3, 4] but without any clear conclusions. In [5] it is explained that clustering is not meaningless but rather difficult due to the fractal auto-similarity of time series. In [6] it is however shown that kernel-based clustering methods are able to provide meaningful clustering.
In this paper, it will be shown that, under certain conditions, the clustering of time series is indeed meaningful. Self-Organizing Maps (SOM) [7] are used here as a clustering tool, thanks to their vector quantization property. Attention will be drawn on the preprocessing that should be applied to time series before creating the regressors. The evidence that clustering of such preprocessed series is meaningful will be provided through simulations on artificial and real time series.

G. Simon is funded by the Belgian F.R.I.A. M. Verleysen is a Senior Research Associate of the Belgian F.N.R.S.

The rest of this paper is organized as follows. Section 2 defines the notations, the algorithm used for clustering, and the criterion that will be used to assess the meaningfulness of time series clustering. This section then presents the correlation problem observed when clustering time series regressors. Section 3 introduces the preprocessing that should be applied in order to unfold the regressor distributions, which decreases the correlation problem. Section 4 presents experimental results on both preprocessed and non-preprocessed time series. Empirical evidence of the meaningfulness of time series clustering will then be provided, as well as the usefulness of an appropriate preprocessing used before creating the regressors. Section 5 briefly concludes this paper.

2 Regressor clustering and its limitations

2.1 Notations and definitions

A time series S is an ordered series of values x_t measured from a process varying in time. The x_t values are usually measured at regular time intervals.
To predict a time series, a model has first to be built. Most of the time this model aims at describing the dependencies between a few past values (given as input) and a single output value (the next value for a given t). In this paper, we consider for simplicity reasons past values only as inputs to the model, excluding exogenous variables. The p-dimensional vector of inputs is called the *regressor* and is built in a sliding window manner as

$$x_{t-p+1}^{t} = \{x_t, x_{t-1}, ..., x_{t-p+1}\}. \quad (1)$$

The question regarding how to chose the length p of the regressor is decisive for the model. For nonlinear models, model selection strategies using statistical resampling methods (cross-validation, k-fold cross-validation, bootstrap, etc. [8]) are usually used to help selecting p, which is related to model complexity and structure selection. The question of choosing an adequate value of p is out of the scope of this paper. In the remaining of this paper, p will therefore be deemed to be fixed a priori.

2.2 Self-Organizing Map algorithm

In this paper, the vector quantization property of Self-Organizing Maps (SOM) [7] is used to perform the clustering of time series regressors. The SOM algorithm is a popular unsupervised classification algorithm that has been applied in many application areas since its introduction in the 80's [7]. The theoretical properties of SOM are now well established [9].
During the learning stage, a SOM moves a fixed number of prototypes inside the data space. The final positions of these prototypes represent a discrete and rough approximation of the data density. The prototypes are linked by neighborhood relations a priori fixed according to a 1- or 2-dimensional grid. The learning consists first in presenting a data, and selecting the winner prototype as the one closest to the given data according to some distance measure. This winning prototype then moves towards the data while the grid indicates which neighbor prototypes may also move. At the end of the learning stage, each prototype is associated with a region of the original space. The data are therefore partitioned into *clusters*. Clustering trough vector quantization is the first property of the SOM. The second property is the topology preservation by which two similar data belong either to the same cluster or to two

neighboring ones (on the grid). The intuitive graphical representations that can be obtained from a SOM make this tool a very popular nonlinear clustering method.

2.3 Comparison criterion

In order to assess the usefulness of time series clustering, we will consider the clustering of various time series. If clustering results on different time series are similar, we will conclude that the clustering method has not been able to capture the specificities of a given regressor distribution and is therefore meaningless. On the contrary, if clustering results on different time series are sufficiently dissimilar, we will conclude that the information contained in the regressors is not lost during clustering, therefore that time series clustering is meaningful.
'Sufficiently dissimilar' means that clusterings performed on various time series lead to much more different results than clusterings performed on the same series, with e.g. different initializations. By 'results of the clustering', it is meant the location of the prototypes: comparing clustering results will therefore necessitate the definition of a criterion that assesses the differences between two sets of prototypes.
By construction, SOM prototypes are located in the regressor space. To compare two sets of prototypes, it is therefore mandatory to scale identically for all series the region of the space covered by the regressors. In the following of this paper, when referring to time series, it is thus meant time series that are normalized according to $x'(t) = (x(t) - \mu_S)/\sigma_S$, where μ_S and σ_S are the time series mean and standard deviation respectively.
Suppose now that two sets A and B of prototypes are obtained from time series S_1 and S_2 using a clustering algorithm, such as the SOM. Of course the regressor length p must be identical for all prototypes in both A and B. Furthermore, to allow a fair comparison, A and B must contain the same number of prototypes I, i.e. $A = \{\bar{a}_i \mid 1 \leq i \leq I\}$ and $B = \{\bar{b}_j \mid 1 \leq j \leq I\}$. To compare these sets of prototypes, the following measure is defined:

$$\textit{position\_difference}(A, B) = \sum_{i=1}^{I} \min_j \left(\text{dist}(\bar{a}_i, \bar{b}_j)\right), \text{ where } 1 \leq j \leq I. \tag{2}$$

This criterion sums, for each prototype in A, the distance to the closest prototype in B. Note that a one-to-one relation between prototype in A and B could be forced by avoiding that the same prototype in B could be selected more than once. However experiments made by this variant do not lead to different conclusions. In the following of this paper, we will thus restrict the results to the *position_difference*(.,.) criterion as defined above.
In practice prototype initialization may considerably influence the clustering results. The clustering is therefore repeated K times on each series, leading to sets of prototype sets:

$$\mathcal{A} = \{A_k \mid 1 \leq k \leq K \text{ and } A_k = \{\bar{a}_i^k \mid 1 \leq i \leq I\}\}, \text{ and} \tag{3}$$

$$\mathcal{B} = \{B_l \mid 1 \leq l \leq K \text{ and } B_l = \{\bar{b}_j^l \mid 1 \leq j \leq I\}\}, \tag{4}$$

To assess the intrinsic difference between several runs of the clustering algorithm on the same series, and similar runs on different series, two criteria are defined:

$$\textit{within}(\mathcal{A}) = \sum_{k_1=1}^{K} \sum_{k_2=1}^{K} \textit{position\_difference}(A_{k_1}, A_{k_2}), \text{ and} \tag{5}$$

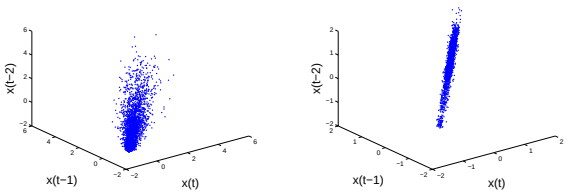

Figure 1: 3-Dimensional regressor distributions obtained from two time series. Left: Sunspot, Right: random walk with Gaussian noise.

$$between(\mathcal{A}, \mathcal{B}) = \sum_{k=1}^{K} \sum_{l=1}^{K} position\_difference(A_k, B_l). \tag{6}$$

The meaningfulness of time series clustering will be assessed by comparing the differences between the two criteria: a large difference will mean that the sets of prototypes resulting from series A and B are *sufficiently dissimilar*, leading to the conclusion of meaningfulness.

2.4 Regressor clustering limitation

When creating the regressors according to Equation (1), care should be taken. Indeed let us imagine a smooth time series, in which variations between successive values are very small. Short regressors built according to (1) will therefore contain very close, or much correlated, values. If all regressors are built similarly, they will be concentrated around a diagonal in the regressor space. This is obviously not a good idea: in this case they will not contain any useful information characterizing the series, and it is not surprising that further clustering will be unable to extract meaningful information from different prototype sets.
Consider the cases of 3-dimensional regressor ($p = 3$) obtained from the Sunspot time series, available from [10], and from a random walk time series generated according to $x(t+1) = x(t) + \varepsilon_t$, where ε_t is a Gaussian-distribution random noise. As this series is then normalized any noise variance can be used. The respective 3-dimensional regressor distributions are shown in Figure 1 left and right respectively. It is obvious that the third component is theoretically not necessary for the noiseless random walk dataset: $p = 2$ is sufficient in that case. However $p = 3$ has been chosen for illustration purposes. In any case, the value of p that will be used in criterion (6) must be identical for the two series, the purpose being to measure the difference between the respective unfoldings as detailed in section 3, even if one of these unfoldings is difficult or impossible in the p-dimensional space.
Figure 1 clearly shows the regressors concentrated around the main diagonal of the 3-dimensional cube. In order to compare series in their regressor spaces, some preprocessing is mandatory. The preprocessing should result in an *unfolding* of the regressor distribution, making the latter more informative about the specificity of each series. Section 3 propose a methodology to implement this preprocessing.

3 Preprocessing methodology

To overcome the limitation illustrated in section 2.4, it is suggested to use a subsampling preprocessing. Indeed the high correlation between successive values in a regressor built

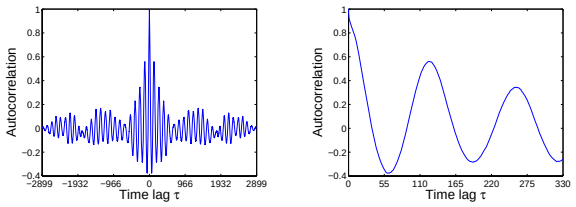

Figure 2: Autocorrelation plot for the Sunspot series. Left: auto-correlation function, right: zoom around the origin (lag $\tau = 0$).

according to Equation (1) may be seen as a too high frequency sampling (with respect to p). A decreasing of the sampling frequency should thus remedy to the problem. Accordingly, it is suggested to build the regressors according to

$$x^t_{t-(p-1)\tau} = \{x_t, x_{t-\tau}, x_{t-2\tau}, ..., x_{t-(p-1)\tau}\}, \tag{7}$$

where τ is a fixed time lag. The difficult part of this subsampling preprocessing is the selection of an adequate value of τ. Note first that several τ values could be adequate: the purpose is to make the regressor distribution span a larger part of the regressor space than the region around the diagonal, but the way how it spans this region is not very important. The purpose is thus to select *any* adequate value of τ, not *the* optimal one according to some criterion. The following procedure, based on the autocorrelation of the time series, is proposed to help selecting this lag.
For discrete real-valued time series, the autocorrelation function $C(\tau)$ is defined as

$$C(\tau) = \sum_{t=1}^{n} x(t)x(t+\tau) \quad \text{for} \quad 0 \leq \tau \leq M, \tag{8}$$

where n is the number of data in the time series and $M \leq n-1$ is the maximum time lag. Function $C(\tau)$ is sometimes called the intrinsic correlation in the literature.
The autocorrelation computed on a time series can then be plotted as a function of the lag. Lag τ^* selected for the subsampling preprocessing is chosen according to the following requirements. First, τ^* should be chosen as small as possible to keep the regressors compact in time. Second, $\mid C(\tau^*) \mid$ should be far from 1, in order to prevent the choice of too correlated values. Third it can be useful to take τ^* such that some of its multiples have (small) negative autocorrelation values. This allows the unfolding to span a larger part of the regressor space. Any reasonable value of τ^* that satisfies these requirements may be chosen. Obviously, as the autocorrelation function are different between time series, the selected value of τ^* will differ too.
This methodology is applied to the Sunspot time series. Figure 2 shows its autocorrelation plot, on the left hand side, and a zoom of this plot near the origin (lag $\tau = 0$), on the right hand side. According to this second figure, the lag $\tau^* = 55$ is chosen. Figure 3 left shows the 3-dimensional regressor distribution obtained using the unfolding preprocessing with $\tau^* = 55$. By comparison, Figure 3 right shows the result of the same procedure applied to the random walk series, where the lag $\tau^* = 100$ has been chosen. A visual inspection of the figures now clearly shows that the distributions are dissimilar, much more than in Figure 1.

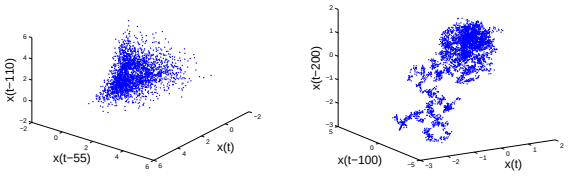

Figure 3: 3-Dimensional plot of lagged regressors obtained using the unfolding preprocessing. Left: Sunspot, $\tau^* = 55$, right: random walk with Gaussian noise, $\tau^* = 100$.

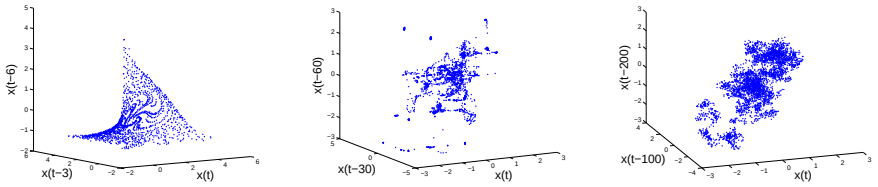

Figure 4: 3-Dimensional plot of lagged regressors. Left: Santa Fe A , $\tau^* = 3$; center: Power plant, $\tau^* = 30$; right: second random walk series, $\tau^* = 100$.

4 Experimental results

In addition to the Sunspot and random walk series used above, the Santa Fe A [11], the Power plant [10] and another random walk time series have also been used for the experiments. The unfolded regressor distributions obtained from these time series using the unfolding preprocessing proposed in section 3 are shown in Figures 4 left, center and right respectively. The SOM algorithm has been run 50 times ($K = 50$) in all experiments below.

First, 3-dimensional regressors obtained from the five considered time series using Equation (1) are clustered. The clustering is performed using 1-dimensional SOM with the number of prototypes varying from 5 to 100 by step of 5. Then the *within(X)* and *between(X, Y)* values are computed, where X is first the Sunspot series and Y either one of the four remaining ones. The computation of the *within(X)* and *between(X, Y)* values is repeated with $X =$ Power plant and with $X =$ Santa Fe A, while Y is again each one of the four remaining series. Then, the whole experiment is repeated using the lagged regressors obtained according to Equation (7). Figure 5 shows the evolution of the *within(.)* and *between(.,.)* values according to the number of prototypes. The three top figures are obtained with non-preprocessed regressors, while the three bottom ones are obtained using the unfolding preprocessing.

First, it can be seen that in all cases, the *between(.,.)* values are above the *within(.)* ones. This confirms the fact that prototype sets resulting from the quantization of different time series differ more than prototype sets resulting from different quantizations of the same time series. In these examples, it is possible to conclude that time series clustering *is* meaningful. A more in-depth discussion should focus on *how much* the *within(.)* and *between(.,.)* criteria differ. Looking for example at the Sunspot case (the two figures on the left), it can be observed that, though all criteria increase more or less linearly with the number of prototypes, the doted lines representing the *between(.,.)* values are much higher in the bottom figure than in the

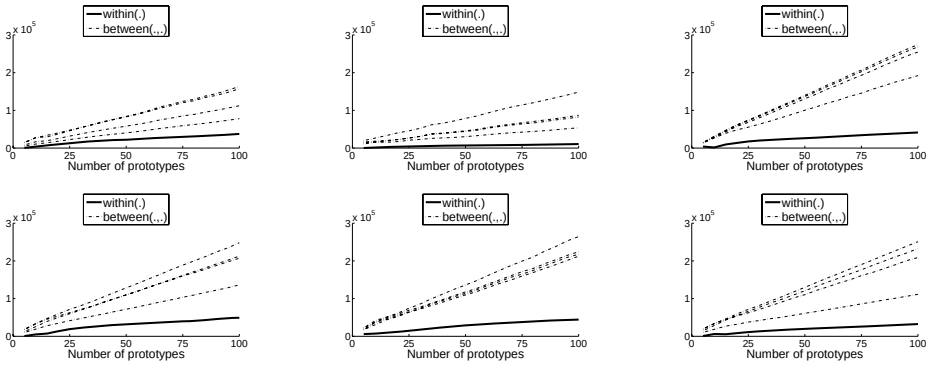

Figure 5: Evolution of the *within(.)* and *between(.,.)* criteria according to the number of prototypes on the SOM (3-dimensional regressor). Top: non-preprocessed regressors, bottom: regressors obtained using the unfolding preprocessing. From left to right: Sunspot, Power plant and Santa Fe A series.

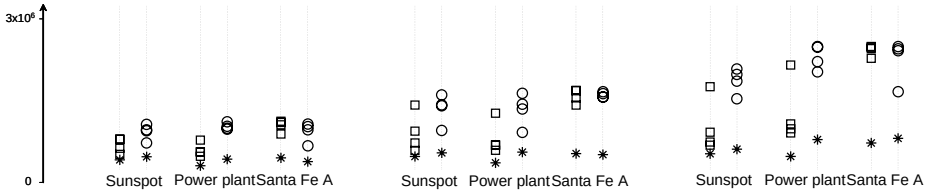

Figure 6: Respective values for $within(.)$ and $between(.,.)$ criteria, for a fixed number of 30 prototypes. Left: $p = 3$; center: $p = 5$; right: $p = 10$. The '*' represent the *within(.)* values, the '□' and '○' represent the *between(.,.)* values for non-preprocessed and unfolded regressors respectively.

top one. This proves that unfolded regressor distributions are more easily distinguishable than non-preprocessed ones. The same observation can be made for the Power plant series and, to a lower extend, for the Santa Fe A time series; this last result is due to the fact that a $\tau^* = 1$ lag already unfolds rather well the Santa Fe A series.
The same experiences are illustrated on regressors of length $p = 5$ and $p = 10$. In Figure 6, it is shown that in these cases the comparisons are even more in favor of the unfolding preprocessing while the regressor length p increases: the gap between the *within(.)* and *between(.,.)* values raises when using preprocessed regressors, and this importantly when p increases. Once again, the Santa Fe A series shows a distinct behavior because it is already unfolded in the regressor space when the lag $\tau^* = 1$; it is thus natural to observe no real difference of *within(.)* and *between(.,.)* values without and with preprocessing in this case.
Finally, let us mention that other clustering algorithms including competitive learning and k-means have also been used to perform the same experiments, confirming the results obtained with SOM and presented in this paper.

5 Conclusion

This paper shows why the clustering of time series regressors might have been considered as meaningless to some extend. However, it is shown that an adequate preprocessing aiming at

unfolding the regressor distribution in the regressor space helps to characterize the specificities of their distribution, including after clustering. With this preprocessing, the clustering becomes clearly meaningful, as illustrated through experiments performed on real and artificial time series. It is also shown that the benefit of this preprocessing increases with the dimension of the regressors.

References

[1] E. Keogh, J. Lin, W. Truppel, (2003), Clustering of Time Series Subsequences is Meaningless: Implications for Past and Future Research, in *Proc. of the 3rd IEEE Int. Conf. on Data Mining*, Melbourne, FL., Nov. 19-22, pp. 115-122.

[2] A. J. Bagnall, G. Janakec, M. Zhang, (2003), Clustering Time Series from Mixture Polynomial Models with Discretised Data, *Technical Report CMP-C03-17*, School of Computing Sciences, University of East Anglia, available from `http://www2.cmp.uea.ac.uk/~ajb/PDF/CMP-C03-17.pdf`.

[3] M. L. Hetland, (2003) Evolving Sequence Rules, *Ph. D. Thesys*, Norwegian University of Computer and Information Science.

[4] M. V. Mahoney, P. K. Chan, (2005) Learning Rules for Time Series Anomaly Detection, *Computer Science Technical Report CS-2005-04*, Computer Sciences Departement, Florida Institute of Technology, Melbourne, FL, available from `http://www.cs.fit.edu/~tr/cs-2005-04.pdf`.

[5] Z. Struzik, (2003), Time Series Rule Discovery: Tough, not Meaningless, in *Proc. of Int. Symp. on Methologies for Intelligent Systems (ISMIS)*, Maebashi City, Japan, Oct. 28-31, Lecture Notes in Artificial Intelligence, **vol. 2871**, Springer-Verlag, pp. 32-39.

[6] A. Denton, (2004), Density-based clustering of time series subsequences, in *Proc. of 3rd Workshop on Mining Temporal and Sequential Data, in conjunction with 10th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, Seattle, WA, Aug. 22-25.

[7] T. Kohonen, (1995), *Self-organising Maps* (2nd ed.), Springer Series in Information Sciences, Berlin, Springer.

[8] B. Efron, R. J. Tibshirani, (1993), *An introduction to the bootstrap*, London, Chapman & Hall.

[9] M. Cottrell, J.-C. Fort, G. Pagès, (1998), Theoretical aspects of the SOM algorithm, *Neurocomputing*, **vol. 21**, pp. 119-138.

[10] E. Keogh, T. Folias, (2004), The UCR time series data mining archive, `http://www.cs.ucr.edu/~eamonn/TSDMA/main.php`.

[11] A. Weigend, N. Gershenfeld, (1994), *Time Series Prediction: Forecasting the future and Understanding the Past*, Santa Fe Institute, MA, Addison-Wesley Publishing Company.