Clustering of time series based on forecasting performance of global models

Ángel López-Oriona[1 [0000−0003−1456−7342]], Pablo Montero-Manso[2 [0000−0003−3816−0985]], and José A. Vilar[1 [0000−0001−5494−171X]]

[1] Research Group MODES, Research Center for Information and Communication Technologies (CITIC), University of A Coruña, 15071 A Coruña, Spain
{a.oriona,jose.vilarf}@udc.es
[2] The University of Sydney Business School, Sydney, Australia
pablo.monteromanso@sydney.edu

Abstract. This article proposes a new procedure to perform clustering of time series. The approach relies on the classical K-means clustering method and is based on two iterative steps: (i) K global forecasting models are fitted via pooling by using the series belonging to each group and (ii) each series is assigned to the cluster associated with the model yielding the best forecasts in accordance with a specific criterion. The resulting clustering solution includes groups which are optimal in terms of overall prediction error, and thus the procedure is able to detect the different forecasting patterns existing in a given dataset. Some simulation experiments show that our method outperforms several alternative techniques in terms of both clustering accuracy and forecasting error. The procedure is also applied to carry out clustering in three real time series databases.

Keywords: time series · clustering · global forecasting models · prediction error · K-means

1 Introduction

Time series clustering (TSC) is a fundamental problem in machine learning with applications in many fields, including geology, finance, computer science or psychology, among others. The task consists of splitting a large collection of unlabelled time series realizations into homogeneous groups so that similar series are located together in the same group and dissimilar series are placed in different clusters. As result, each group can be characterized by a specific temporal pattern, which allows to address key issues as discovering hidden dynamic structures, identifying anomalies or forecasting future behaviours. A comprehensive overview on the topic is provided in [10].

A crucial point in cluster analysis is to establish the dissimilarity notion since it determines the nature of the resulting clustering partition. Several distance measures have been proposed in the literature, each one of them associated with a different objective. If the goal is to discriminate between geometric profiles of

the time series, then a shape-based dissimilarity is suitable. For instance, the well-known dynamic time warping (DTW) distance has been used in several works to perform TSC [3]. On the contrary, a structure-based dissimilarity is desirable if the target is to compare underlying dependence models. Examples of this type of distances are metrics comparing the autocorrelations [5] or the wavelet coefficients [4] of two time series. Additional types of dissimilarities are based on estimated model coefficients [2].

The goal of this work is to construct a TSC algorithm capable of returning a partition which is optimal in terms of overall forecasting accuracy. To that aim, we introduce the notion of dissimilarity between a time series and a given model (e.g., ARIMA) as the average prediction error produced when iteratively obtaining the point forecasts of the time series with respect to the corresponding model. It is worth highlighting that, although there are a few TSC methods based on forecast densities [14], to the best of our knowledge, nobody has employed the concept of similarity previously exposed to perform clustering in time series databases. Specifically, our clustering approach makes use of the so-called global models to minimize the average prediction error. Global models are constructed in the following way [12]: (i) each series in a set is lag-embedded into a matrix at a given AR order, l, fixed beforehand, (ii) these matrices are stacked together to form one big matrix, achieving data pooling and (iii) a classical regression model (e.g., linear regression, random forest etc) is fitted to the resulting matrix.

Global models have been shown to outperform local models in terms of forecasting accuracy in several datasets [12]. In other words, when a single model is fitted to all the time series in the database, and used to obtain the corresponding predictions, a lower average forecasting error is produced than in the case where each time series is predicted by considering a different local model. Moreover, global models do not need any assumption about similarity of the time series in the collection, and need far fewer parameters than the simplest of local methods.

Although the global model approach produces outstanding results, it has one important drawback: it ignores the possible existence of homogeneous groups of series in terms of prediction patterns. For instance, a database could contain two groups of series in such a way that the series within each group are helpful to each other for obtaining accurate predictions (e.g., think of several countries whose behaviour concerning monthly economic growth is very similar), but totally useless for the series in the remaining group. In the previous situation, it would be desirable to fit a global method for each distinct set of time series. Then the predictions would be computed for a given series by using its associated global model. In order to detect groups of series sharing similar forecasting structures, we propose a novel clustering method which is based on the traditional K-means algorithm. The technique relies on the following iterative process: (i) K global models (centroids) are fitted by taking into account the series pertaining to each cluster independently and (ii) each time series is assigned to the group associated with the centroid producing the lowest forecasting error according to a specific metric.

It is worth emphasizing that, by construction, the proposed algorithm produces a partition which is optimal in terms of overall prediction effectiveness. In fact, the objective function of the pseudo K-means method can be seen as a sum of forecasting errors (see Section 2), which is expected to decrease with each iteration of the two-step procedure described above. Therefore, the clustering algorithm is specifically designed to allocate the different time series in such a way that the corresponding global models represent in the best possible manner the existing prediction patterns. There are only a few works in the literature combining clustering and global methods in a single technique. For instance, [1] proposed an approach particularly devised to improve the forecasting accuracy of global models. First, the set of series is partitioned into different groups by using a specific clustering method. Then, global models are fitted by considering the series within each cluster. Although successful, the method of [1] splits the set of series by using a feature-based TSC clustering method so that there is not guarantee that the resulting partition is optimal in terms of total prediction accuracy. Note that our approach circumvents this limitation by adapting the objective function to the specific purpose of forecasting error reduction.

Some simulation experiments are carried out in the paper to assess the performance of the proposed algorithm in terms of both clustering effectiveness and forecasting accuracy. In all cases, synthetic partitions where the groups are characterized by different generating processes are considered, and the approach is compared with several alternative methods, as one procedure based on local models or the technique of [1]. The method is also applied to perform clustering in some well-known datasets. Overall, the algorithm exhibits a great behaviour when dealing with both synthetic and real data.

The remainder of this paper is organized as follows. Section 2 describes the clustering algorithm based on prediction accuracy of global forecasting models. The approach is analysed in Section 3 by means of a simulation study where different scenarios are taken into account. In Section 4, we apply the proposed method to real datasets of time series. Section 5 contains some concluding remarks.

2 A clustering algorithm based on prediction accuracy of global forecasting models

Consider a set of n time series, $\mathcal{S} = \left\{\boldsymbol{X}_t^{(1)}, \ldots, \boldsymbol{X}_t^{(n)}\right\}$, where each $\boldsymbol{X}_t^{(i)} = \left(X_1^{(i)}, \ldots, X_{L_i}^{(i)}\right)$ is a series of length L_i, $i = 1, \ldots, n$. We assume that each series $\boldsymbol{X}_t^{(i)}$ contains training and a validation periods of lengths $r(i)$ and $s(i)$, denoted by $\boldsymbol{\mathcal{T}}^{(i)} = (t_1^i, \ldots, t_{r(i)}^i)$ and $\boldsymbol{\mathcal{V}}^{(i)} = (v_1^i, \ldots, v_{s(i)}^i)$, respectively, such that

- Both $\boldsymbol{\mathcal{T}}^{(i)}$ and $\boldsymbol{\mathcal{V}}^{(i)}$ are formed by consecutive observations and t_1^i has a position equal to or less than the position of v_1^i, considering both t_1^i and v_1^i as elements of the vector $\boldsymbol{X}_t^{(i)}$,

- $mset(\boldsymbol{\mathcal{T}}^{(i)}) \subseteq mset(\boldsymbol{X}_t^{(i)})$ and $mset(\boldsymbol{\mathcal{V}}^{(i)}) \subseteq mset(\boldsymbol{X}_t^{(i)})$ (both periods are included in the original series),
- $mset(\boldsymbol{X}_t^{(i)}) \subseteq mset(\boldsymbol{\mathcal{T}}^{(i)}) + mset(\boldsymbol{\mathcal{V}}^{(i)})$ (both periods form a cover of the original series),

where the operator $mset(\boldsymbol{z}) = [z_1, \ldots, z_n]$ for the vector $\boldsymbol{z} = (z_1, \ldots, z_n)$, denoting $[\cdot]$ a multiset, i.e., a generalization of the traditional set in which each element can appear multiple times. Note that, by virtue of the previous three conditions, the training and validation periods may contain common observations. This general feature allows to consider traditional validation measures as the in-sample error.

The sets $\mathcal{T} = \{\boldsymbol{\mathcal{T}}^{(1)}, \ldots, \boldsymbol{\mathcal{T}}^{(n)}\}$ and $\mathcal{V} = \{\boldsymbol{\mathcal{V}}^{(1)}, \ldots, \boldsymbol{\mathcal{V}}^{(n)}\}$ are called the training and the validation set, respectively. We wish to perform clustering on the elements of $\mathcal{S}$ in such a way that the groups are associated with global models minimizing the overall forecasting error with respect to the validation set. The method we propose is a K-means-based algorithm having the classical two stages: (i) constructing a prototype for each cluster, usually referred to as centroid and (ii) assigning every series to a group. The assignment step often relies on the distance from the series to the prototypes. In this work, we propose to consider global models as prototypes for each group. Specifically, the prototype of kth cluster is a global model which is fitted to the series pertaining to kth cluster.

Assume there are n_k series in the kth cluster C_k, i.e., $C_k = \left\{\boldsymbol{X}_{t,k}^{(1)}, \ldots, \boldsymbol{X}_{t,k}^{(n_k)}\right\}$, with $k = 1, \ldots, K$. A global model $\mathcal{M}_k$ is fitted in cluster C_k by considering the training periods associated to $\boldsymbol{X}_{t,k}^{(j)}$, $j = 1, \ldots, n_k$. It is expected that the predictive ability of model $\mathcal{M}_k$ with respect to the series in cluster C_k is better the more related the series in the group are. In sum, the set of clusters $\boldsymbol{C} = \{C_1, \ldots, C_K\}$ produce the prototypes $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_K\}$.

Once the global models $\mathcal{M}_1, \ldots, \mathcal{M}_K$ have been constructed, each series is assigned to the cluster whose prototype gives rise to the minimal value for the mean absolute error (MAE) by considering the validation period. Specifically, series $\boldsymbol{X}_t^{(i)}$, $i = 1, \ldots, n$, is assigned to cluster k' such that

$$k' = \underset{k=1,\ldots,K}{\arg\min}\, d_{\text{MAE}}\big(\boldsymbol{X}_t^{(i)}, \mathcal{M}_k\big) = \underset{k=1,\ldots,K}{\arg\min}\, \frac{1}{s(i)} \sum_{j=1}^{s(i)} \big|v_j^i - F_{j,k}^{(i)}\big|, \tag{1}$$

where $F_{j,k}^{(i)}$ is the prediction of v_j^i by considering the global model $\mathcal{M}_k$. Note that considering the MAE in (1) is appropriate because we are evaluating the forecasting effectiveness of K global models with respect to the ith series independently. Therefore, each assignation is only influenced by the units of the corresponding series so that no scaling issues arise. In fact, the simplicity of the MAE makes it a recommended error metric for assessing accuracy on a single series [9].

Both steps the computation of prototypes and the reassignation of series are iterated until convergence or a maximum number of iterations is reached. The

corresponding clustering algorithm is described in Algorithm1. Below we provide some remarks concerning the proposed method.

Algorithm 1 pseudo-K-means clustering algorithm based on prediction accuracy of global forecasting models

1: Fix K, l and $max.iter$
2: Set $iter = 1$
3: Randomly divide the n series into K clusters
4: Compute the initial set of l-lagged global models $\boldsymbol{\mathcal{M}} = \{\mathcal{M}_1, \ldots, \mathcal{M}_K\} = \boldsymbol{\mathcal{M}}^{(1)}$
5: **repeat**
6: Set $\boldsymbol{\mathcal{M}}_{\text{OLD}} = \boldsymbol{\mathcal{M}}$ {Store the current prototypes}
7: Assign each series to the cluster associated with its nearest prototype according to the rule in (1)
8: Compute the new collection of prototypes by fitting a l-lagged global model to the training periods of the series in kth cluster, $k = 1, \ldots, K$. {Update the set of prototypes}
9: $iter \leftarrow iter + 1$
10: **until** $\boldsymbol{\mathcal{M}} = \boldsymbol{\mathcal{M}}_{\text{OLD}}$ or $iter = max.iter$

Remark 1 (*Interpretation of objective function*). Note that the objective function in Algorithm 1 can be written as

$$J(\boldsymbol{C}) = \sum_{k=1}^{K} \sum_{\substack{i=1: \\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} d_{MAE}(\boldsymbol{X}_t^{(i)}, \mathcal{M}_k), \tag{2}$$

which is a sum of prediction errors with respect to the validation periods. In particular, each series is forecasted by using the global model associated with the cluster it pertains. In this regard, the value of the objective function returned when Algorithm 1 stops, say J_{OPT}, can be regarded as the total optimal (minimal) prediction error when K groups are assumed to exist in the dataset. In the same way, the quantity J_{OPT}/n can be interpreted as the average optimal prediction error. In sum, the objective function of the proposed K-means clustering algorithm is very interpretable from a forecasting perspective.

Remark 2 (*Assessment of the resulting partition in terms of prediction error*). Although the quantity J_{OPT}/n can be seen as the average optimal prediction error, this value is not an appropriate metric to assess the predictive ability of the resulting clustering partition. Note that the two-step procedure described in Algorithm 1 attempts to find the partition minimizing the average prediction error with respect to the validation periods. Therefore, J_{OPT}/n is likely to underestimate the prediction error computed over future periods of the series which are not involved in the optimization process. In this regard, a proper error metric could be obtained through the following steps:

1. Given a prediction horizon $h \in \mathbb{N}$, divide each series into two periods. The first period contains all but the last h observations of the series. The second period, referred to as test period, contains the last h observations. The first periods constitute the set $\mathcal{S} = \left\{ \boldsymbol{X}_t^{(1)}, \ldots, \boldsymbol{X}_t^{(n)} \right\}$ introduced above, whereas the second periods constitute the set $\mathcal{S}^* = \left\{ \boldsymbol{X}_t^{(1)*}, \ldots, \boldsymbol{X}_t^{(n)*} \right\}$, where each $\boldsymbol{X}_t^{(i)*} = (X_1^{(i)*}, \ldots, X_h^{(i)*})$ is a series of length h. The set $\mathcal{S}^*$ is called the test set.
2. Run Algorithm 1 using the set $\mathcal{S}$ as input, obtaining the clustering solution.
3. Given the clustering solution computed in Step 2, and for $k = 1, \ldots, K$, fit a l-lagged global model to the set of series in the kth cluster by considering both training and validation periods. This produces the set of global models $\overline{\boldsymbol{\mathcal{M}}} = \{\overline{\mathcal{M}}_1, \ldots, \overline{\mathcal{M}}_K\}$.
4. Compute the average prediction error with respect to the test set as

$$\frac{1}{n} \sum_{k=1}^{K} \sum_{\substack{i=1: \\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} d^*\big(\boldsymbol{X}_t^{(i)*}, \overline{\mathcal{M}}_k\big), \tag{3}$$

where d^* is any function measuring discrepancy between the actual values of $\boldsymbol{X}_t^{(i)*}$ and their predictions according to model $\overline{\mathcal{M}}_k$. Note that these predictions are computed starting from the series $\boldsymbol{X}_t^{(i)}$ and in a recursive manner. As an example, if the MAE is chosen as the error metric, then (3) becomes

$$\frac{1}{n} \sum_{k=1}^{K} \sum_{\substack{i=1: \\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} d^*_{\mathrm{MAE}}\big(\boldsymbol{X}_t^{(i)*}, \overline{\mathcal{M}}_k\big) = \frac{1}{n} \sum_{k=1}^{K} \sum_{\substack{i=1: \\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} \frac{1}{h} \sum_{j=1}^{h} \big| X_j^{(i)*} - \overline{F}_{j,k}^{(i)*} \big|, \tag{4}$$

where $\overline{F}_{j,k}^{(i)*}$ is the prediction of $X_j^{(i)*}$ according to the global model $\overline{\mathcal{M}}_k$. The R code used for the implementation of Algorithm (1) is available at https://anloor//clustering_procedure.

3 Simulation study

In this section we carry out a set of simulations with the aim of assessing the performance of the proposed approach in different scenarios. Firstly we describe the simulation mechanism, then we explain how the evaluation of the method was done and finally we show the results of the simulation study

3.1 Experimental design

Two specific scenarios were constructed, both of them including linear processes. Specifically, the first and second scenario involve short memory and long memory models, respectively. In this way, the proposed method is analysed under

different degrees of serial dependence. Both scenarios contain three distinct generating processes. The particular generating models are given below.

Scenario 1. Consider the AR(p) process given by

$$X_t = \sum_{i=1}^{p} \varphi_i X_{t-i} + \epsilon_t, \tag{5}$$

where ϵ_t is a process formed by independent elements following the standard Gaussian distribution. We fix $p = 4$. The vector of coefficients $\boldsymbol{\varphi}_4 = (\varphi_1, \varphi_2, \varphi_3, \varphi_4)$ is set as indicated below.

Process 1: $\boldsymbol{\varphi}_4 = (0.1, 0.2, -0.4, 0.3)$.
Process 2: $\boldsymbol{\varphi}_4 = (0.2, -0.5, 0.3, -0.3)$.
Process 3: $\boldsymbol{\varphi}_4 = (-0.3, 0.4, 0.6, -0.2)$.

Scenario 2. Consider the AR(p) process given in (5). We fix $p = 12$. The vector of coefficients $\boldsymbol{\varphi}_{12} = (\varphi_1, \varphi_2, \ldots, \varphi_{12})$ is set as

$(0.9, -0.5, -0.3, 0.3, 0.1, -0.3, 0.2, -0.3, 0.5, -0.5, 0.3, -0.3)$,
$(0.2, 0.3, -0.2, -0.2, 0.4, 0.2, -0.1, 0.2, 0.1, -0.2, -0.3, 0.5)$,
$(-0.3, -0.1, 0.3, -0.1, -0.2, -0.1, -0.4, -0.2, -0.3, 0.4, 0.1, 0.2)$,

for Processes 1, 2 and 3, respectively. It is worth emphasizing that, in both Scenarios 1 and 2, the vectors of coefficients were randomly selected with the only requirement of fulfilling the standard stationary condition for AR processes.

The simulation study was carried out as follows. For each scenario, N time series of length T were generated from each process. Several values of N and T were taken into account to analyse the effect of those parameters (see Section 3.3). The test set was constructed by considering the last $h = 2l_{\text{SIG}}$ observations of each series, where l_{SIG} is the number of significant lags existing in each scenario (e.g., $l_{\text{SIG}} = 4$ in Scenario 1). The training period was set to the first $(T - h)$ observations of each series. The validation period was set to observations from $(l + 1)$ to $(T - h)$. Note that this choice implies that the reassignation step in Algorithm 1 is carried out by considering the in-sample error (see (1)). The simulation procedure was repeated 200 times for each pair (T, N).

3.2 Alternative approaches and assessment criteria

To throw light on the behaviour of the proposed algorithm, which we will refer to as *Clustering based on Prediction Accuracy of Global Models* (CPAGM), we decided to compare it with the alternative approaches described below.

- *Local Models* (LM). Specifically, a local model (e.g., an AR model) is fitted to each of the series in the collection (by jointly considering training and validation periods) and used to obtain the predictions with respect to the test

period. In this way, each local model gives rise to an error metric measuring its forecasting accuracy. The average of these quantities can be seen as the overall error associated to the LM approach. Note that the LM method was already used by [12] to show the benefits of global models for forecasting purposes.

- *Global Models by considering an Arbitrary Partition* (GMAP). This procedure is based on 2 steps: (i) the original set of series $\mathcal{S}$ is randomly partitioned into K groups and (ii) for each group, a global model is fitted by considering the series pertaining to that cluster. The assessment task is carried out as indicated in Step 4 of Remark 2. It is worth highlighting that global models fitted to random groups of series have been shown to improve the forecasting accuracy of one global model fitted to all the series in some datasets (see, e.g., Figure 4 in [12]). The approach GMAP can be seen as a meaningful benchmark for the proposed method, since it is expected that the groups produced by Algorithm 1 improve the forecasting effectiveness of global models in comparison with a random partition.
- *Global models by considering Feature-Based Clustering* (GMFBC). Particularly, the technique proposed by [1], which relies on two steps: (i) the original collection of series is splitted into K groups by using a clustering algorithm based on the feature extraction procedure described in [8] and (ii) K global models are constructed according to the resulting partition. This approach is evaluated in a similar way that GMAP. Note that, like CPAGM, GMFBC also tries to exploit the notion of similarity between time series in order to decrease the overall prediction error. However, GMFBC considers a specific clustering algorithm before fitting the global models, while CPAGM iterates until achieving the optimal clustering partition in terms of forecasting effectiveness.

The number of clusters was set to $K = 3$, since all scenarios contain 3 different generating processes. For approaches CPAGM, GMAP and GMFBC, the number of lags l to fit the global models was set to $l = l_{\text{SIG}}$. The considered global models were standard linear regression models adjusted by least squares. As for the method LM, a linear local model was fitted to each series by using the function **auto.arima** in the forecast R package [7], which contains classical linear regression as a particular case. Model selection was performed by means of AICc criterion. Note that classical linear models are important as a benchmark because they do not include any advanced machine learning technique and overlap the model class with ARIMA model (a common local approach). Therefore, they are ideal to isolate the effect of globality [12].

The quality of the procedures was evaluated by comparing the clustering solution given by the algorithms with the true partition, usually referred to as ground truth. Approaches CPAGM and GMFBC automatically provide a clustering partition. For method LM, each series was first described by means of the vector of estimated model coefficients returned by **auto.arima** function (all vectors were padded with zeros until reaching the length of the longest vector). Next, a standard K-means algorithm was executed by using these feature vectors

as input. Experimental and true partitions were compared by considering the adjusted Rand index (ARI) [6], which is bounded between -1 and 1. Values of ARI close to 0 indicate a noninformative clustering solution, while the closer to 1 the index, the better the partition.

Note that, although CPAGM is a clustering method, it could be used as a tool to perform forecasting in a given set of series, since the iterative process outlined in Algorithm (1) attempts to minimize the total prediction error. In this regard, the forecasting accuracy of methods CPAGM, GMAP and GMFBC was assessed by recording the average MAE as indicated in (4). The MAE associated with each local model computed with respect to the test set was stored for LM and the average of those quantities was calculated as the error metric. Note that, since all series within a given scenario are measured in the same scale, the MAE is a proper measure to evaluate the overall prediction error.

In each simulation trial and given a pair (N, T), the proposed technique CPAGM was executed 5 times and the partition associated with the minimum value of $J(\boldsymbol{C})$ (see (2) in Remark 1) was stored. This way, we tried to avoid the well-known issue of local optima related to K-means-based procedures. A similar strategy was employed in the feature-based clustering of GMAP and GMFBC. The overall MAE produced by GMAP was approximated via Monte Carlo (i.e., by considering several random partitions).

3.3 Results and discussion

Average values of ARI and MAE attained by the different techniques in Scenario 1 are provided in Tables 1 and 2, respectively. In order to perform rigorous comparisons, pairwise paired t-tests were carried out by taking into account the 200 simulation trials. In all cases, the alternative hypotheses stated that the mean ARI (MAE) value of a given method is greater (less) than the mean ARI (MAE) value of its counterpart. As asterisk was incorporated in Tables 1 and 2 if the corresponding method resulted significantly more effective than the remaining ones for a significance level 0.01. The results associated with running the approach CPAGM with $K = 1$ (only one global model) were incorporated to Table 2 by indicating "$(K = 1)$".

According to Table 1, the proposed method CPAGM achieved significantly greater ARI values than the alternative approaches in most of the cases. The only exceptions were $(T, N) = (400, 5)$ and $(T, N) = (400, 20)$, where CPAGM and LM showed a similar performance. What happens here is that, as long series are considered, the models coefficients are very accurately estimated and the clustering partition returned by the LM approach is quite similar to the ground truth. An increasing in the number of series per cluster was clearly beneficial for the proposed method when short series were considered ($T \in \{20, 50\}$), but it had little impact when $T > 50$. In some way, more series per cluster has a similar effect on CPAGM than longer lengths, since both phenomena result in better estimated global models. The approach GMFBC showed a steady improvement when increasing the series length, but it was still far from a perfect partition for $T = 400$. Due to a reviewer's suggestion, Table 1 also contains the results

(T, N)	LM	CPAGM	GMFBC	DTW	KS
(20, 5)	0.027	**0.352***	0.094	-0.016	0.308
(20, 10)	0.032	**0.459***	0.090	0.023	0.270
(20, 20)	0.029	**0.556***	0.092	0.004	0.254
(20, 50)	0.026	**0.612***	0.076	0.019	0.308
(50, 5)	0.305	**0.914***	0.243	0.031	0.625
(50, 10)	0.336	**0.956***	0.222	0.021	0.777
(50, 20)	0.331	**0.988***	0.216	0.016	0.758
(50, 50)	0.331	**0.981***	0.195	0.026	0.864
(100, 5)	0.747	**0.946***	0.379	0.042	0.717
(100, 10)	0.740	**0.954***	0.380	0.028	0.870
(100, 20)	0.743	**0.961***	0.334	0.026	0.818
(100, 50)	0.740	**0.956***	0.311	0.025	0.799
(200, 5)	0.876	**0.906***	0.581	0.046	0.831
(200, 10)	0.854	**0.919**	0.561	0.040	0.873
(200, 20)	0.820	**0.921***	0.516	0.025	0.813
(200, 50)	0.800	**0.926***	0.488	0.030	0.817
(400, 5)	0.897	**0.908**	0.719	0.010	0.825
(400, 10)	0.848	**0.900***	0.725	0.022	0.769
(400, 20)	0.877	**0.881**	0.732	0.036	0.841
(400, 50)	0.803	**0.872***	0.726	0.034	0.688

Table 1. Average ARI in Scenario 1. The best result is shown in bold. An asterisk indicates that a given method is better than the rest at level 0.01.

(T, N)	LM	CPAGM ($K = 1$)	GMFBC	GMAP
(20, 5)	1.066	**1.043*** (1.069)	1.072	1.078
(20, 10)	1.068	**0.997*** (1.075)	1.046	1.080
(20, 20)	1.070	**0.964*** (1.076)	1.036	1.052
(20, 50)	1.073	**0.942*** (1.075)	1.034	1.046
(50, 5)	1.019	**0.921*** (1.065)	1.011	1.100
(50, 10)	1.023	**0.913*** (1.073)	1.021	1.044
(50, 20)	1.024	**0.910*** (1.082)	1.024	1.072
(50, 50)	1.016	**0.907*** (1.074)	1.020	1.042
(100, 5)	0.976	**0.919*** (1.072)	0.994	1.225
(100, 10)	0.978	**0.913*** (1.075)	0.996	1.148
(100, 20)	0.976	**0.911*** (1.076)	1.003	1.067
(100, 50)	0.977	**0.911*** (1.079)	1.009	1.061
(200, 5)	0.929	**0.911*** (1.062)	0.949	1.025
(200, 10)	0.942	**0.918*** (1.083)	0.968	1.058
(200, 20)	0.938	**0.912*** (1.070)	0.969	1.062
(200, 50)	0.942	**0.916*** (1.073)	0.978	1.090
(400, 5)	0.920	**0.915** (1.069)	0.937	1.092
(400, 10)	0.920	**0.916** (1.076)	0.937	1.069
(400, 20)	0.929	**0.926** (1.080)	0.949	1.071
(400, 50)	0.925	**0.925** (1.076)	0.944	1.101

Table 2. Average MAE in Scenario 1. The best result is shown in bold. An asterisk indicates that a given method is better than the rest at level 0.01.

associated with a standard K-means approach based on the DTW distance and with the method of [13], denoted by KS, which relies on a normalized version of the cross-correlation. Both methods exhibited worse overall behaviour than the proposed approach, although KS obtained large values for the ARI index in many settings.

The results in terms of MAE (see Table 2) are very similar to those in Table 1, with the proposed method outperforming the remaining approaches in most of the settings. Specifically, Table 2 indicates that the forecasting accuracy of local models is as good as that of global models for $T = 400$, but significantly worse for shorter lengths. Note that CPAGM obtained substantially better results than fitting one global model to all the series in the collection ($K = 1$) and GMAP, which is expected since these approaches do not take into account the underlying generating processes.

Average results for Scenario 2 concerning ARI and MAE are displayed in Tables 3 and 4, respectively. The proposed approach showed a similar behaviour than in Scenario 1 in terms of both clustering effectiveness and predictive accuracy, but the difference with respect to the remaining techniques was more marked in Scenario 2. The long memory patterns arising in the processes of this scenario negatively affected both methods LM and GMFBC. In fact, the LM approach was not able to exhibit forecasting and clustering accuracies similar to CPAGM even when very long series ($T = 1000$) were considered. Method KS displayed a similar performance than CPAGM in this scenario. In short, the iterative procedure of Algorithm 1 takes advantage of the excellent accuracy of global models to properly estimate the complex forecasting patterns arising in the long-memory processes of Scenario 2.

(T, N)	LM	CPAGM	GMFBC	DTW	KS
$(50, 5)$	0.243	0.584	0.238	0.107	**0.765***
$(50, 10)$	0.259	**0.853***	0.222	0.135	0.789
$(50, 20)$	0.250	**0.956***	0.219	0.144	0.723
$(50, 50)$	0.256	**0.980***	0.205	0.155	0.767
$(100, 5)$	0.386	**0.933***	0.278	0.198	0.869
$(100, 10)$	0.387	**0.937***	0.274	0.122	0.907
$(100, 20)$	0.410	**0.979**	0.277	0.144	0.968
$(100, 50)$	0.412	**0.986***	0.286	0.158	0.927
$(200, 5)$	0.453	0.907	0.302	0.123	**0.934**
$(200, 10)$	0.478	**0.937***	0.317	0.174	0.897
$(200, 20)$	0.468	**0.959**	0.306	0.152	0.937
$(200, 50)$	0.477	**0.972***	0.303	0.128	0.920
$(400, 5)$	0.517	0.898	0.383	0.165	**0.955***
$(400, 10)$	0.510	0.918	0.382	0.160	**0.921**
$(400, 20)$	0.507	**0.926***	0.368	0.169	0.883
$(400, 50)$	0.487	0.921	0.365	0.131	**0.983***
$(1000, 5)$	0.571	0.846	0.497	0.184	**0.935***
$(1000, 10)$	0.556	0.841	0.456	0.249	**0.878**
$(1000, 20)$	0.552	**0.867**	0.453	0.272	0.863
$(1000, 50)$	0.532	0.877	0.457	0.273	**0.923***

Table 3. Average ARI in Scenario 2. The best result is shown in bold. An asterisk indicates that a given method is better than the rest at level 0.01.

(T, N)	LM	CPAGM ($K = 1$)	GMFBC	GMAP
$(50, 5)$	1.854	**1.375*** (1.871)	1.657	1.902
$(50, 10)$	1.855	**1.333*** (1.885)	1.616	1.888
$(50, 20)$	1.856	**1.183*** (1.905)	1.625	1.838
$(50, 50)$	1.857	**1.153*** (1.901)	1.647	1.898
$(100, 5)$	1.670	**1.185*** (1.871)	1.492	1.756
$(100, 10)$	1.665	**1.173*** (1.891)	1.553	1.667
$(100, 20)$	1.683	**1.148*** (1.898)	1.578	1.890
$(100, 50)$	1.683	**1.147*** (1.903)	1.590	1.884
$(200, 5)$	1.615	**1.191*** (1.884)	1.507	1.613
$(200, 10)$	1.628	**1.168*** (1.899)	1.558	1.772
$(200, 20)$	1.635	**1.156*** (1.902)	1.591	1.852
$(200, 50)$	1.631	**1.152*** (1.906)	1.624	1.866
$(400, 5)$	1.566	**1.197*** (1.906)	1.483	1.743
$(400, 10)$	1.574	**1.177*** (1.898)	1.526	1.729
$(400, 20)$	1.561	**1.177*** (1.900)	1.573	1.885
$(400, 50)$	1.563	**1.181*** (1.904)	1.596	1.916
$(1000, 5)$	1.486	**1.219*** (1.885)	1.394	1.898
$(1000, 10)$	1.513	**1.231*** (1.899)	1.473	1.887
$(1000, 20)$	1.516	**1.210*** (1.908)	1.497	1.892
$(1000, 50)$	1.505	**1.205*** (1.902)	1.516	1.881

Table 4. Average MAE in Scenario 2. The best result is shown in bold. An asterisk indicates that a given method is better than the rest at level 0.01.

4 Application to real data

In this section we apply the proposed algorithm to perform clustering in 3 well-known datasets. They have been used in many peer-reviewed publications as standard benchmarks, from local models to recent literature for global models. Specifically, [12] employed these databases to show the advantages of global methods over local methods in terms of forecasting accuracy. The datasets pertain in turn to the data collection M1, used in a forecasting competition [11]. It contains 1001 series subdivided in yearly (181), quarterly (203) and monthly (617) periodicity. These three subsets define precisely the three considered databases.

Method CPAGM and the alternative approaches studied in Section 3 were executed in each of the three datasets. No data preprocessing was performed, since there is not a clear agreement about the benefits of preprocessing when fitting global models [12]. Note that, unlike in the simulation study, there is no way of objectively assessing the quality of the clustering partition in these databases, since no information about the ground truth is available. Hence, our analyses focus on the predictive effectiveness of the evaluated techniques. Procedures CPAGM, GMFBC and GMAP were run for several values of K, namely $K \in \{1, 2, 3, 4, 5, 7, 10\}$ and l. Note that the range of l is limited by the minimum series length existing in a given database.

To measure the forecasting accuracy, we considered the symmetric Mean Absolute Percentage Error (sMAPE). Using a percentage error is desirable here because, unlike in the numerical experiments of Section 3, some databases contain series which are recorded in very different scales. Thus, employing the MAE

could have resulted in the average forecasting error being corrupted by the higher influence of the series in the largest scales. Note that, by considering the sMAPE metric, the average prediction error in (3) takes the form

$$\frac{1}{n}\sum_{k=1}^{K}\sum_{\substack{i=1:\\ \boldsymbol{X}_t^{(i)}\in C_k}}^{n} d^*_{\text{sMAPE}}\left(\boldsymbol{X}_t^{(i)*},\overline{\mathcal{M}}_k\right)=\frac{1}{n}\sum_{k=1}^{K}\sum_{\substack{i=1:\\ \boldsymbol{X}_t^{(i)}\in C_k}}^{n}\frac{200}{h}\sum_{j=1}^{h}\left(\frac{\left|X_j^{(i)*}-\overline{F}_{j,k}^{(i)*}\right|}{\left|X_j^{(i)*}\right|+\left|\overline{F}_{j,k}^{(i)*}\right|}\right). \tag{6}$$

Concerning the proposed algorithm CPAGM, the test sets were constructed by considering the last $h = 5$ observations of each time series. As in the simulations of Section 3, the in-sample error was used to assign each series to its closest cluster in the iterative mechanism of Algorithm 1. Traditional least squares linear regression was considered to fit the global models.

Figures 1, 2 and 3 contain the results for yearly, quarterly and monthly datasets, respectively. Left, middle and right panels refer to the approaches CPAGM, GMFBC and GMAP, respectively. Curves of average sMAPE are depicted as a function of the number of lags, l. Each colour corresponds to a different value of the number of clusters, K. In all cases, the proposed algorithm CPAGM attains a substantially lower average error than the alternative methods GMFBC and GMAP for a large number of pairs (K, l). Specifically, the differences by taking into account the minimum average errors (those associated with the optimal pair for each method) are dramatic in some cases. For instance, in dataset M1 Quarterly, procedures GMFBC and GMAP obtain a minimum average error two times and three times higher, respectively, than the one associated with CPAGM. In addition, considering a number of clusters of $K > 1$ is advantageous in the three datasets, as the red curve is significantly above the remaining curves in all of the settings. This suggests that M1 databases contain groups of series sharing common forecasting patterns, and thus fitting a global model to the series within each group is beneficial in terms of forecasting effectiveness.

It is worth highlighting that, for the proposed approach, there is usually at least one nonoptimal pair (K, l) for which the average sMAPE is not significantly different from that of the optimal one. For example, in dataset M1 Quarterly, pairs $(10, 10)$ and $(7, 6)$ produce almost the same average error. In such a case, selecting the latter could be more appropriate for performing further data mining tasks, as it would result in better interpretability of centroids (fewer parameters) and lower computational complexity.

Optimal pairs (K, l) for each method in Figures 1, 2 and 3 are summarized in Table 5 along with the corresponding forecasting errors. Average sMAPE attained by the local-based approach LM in each dataset is also shown. It is clear that the method CPAGM outperforms all alternative approaches in the three cases.

In short, the application of this section shows the advantages of the proposed algorithm CPAGM when performing forecasting in time series databases. It is worth highlighting that the proposed algorithm was also applied to perform clus-

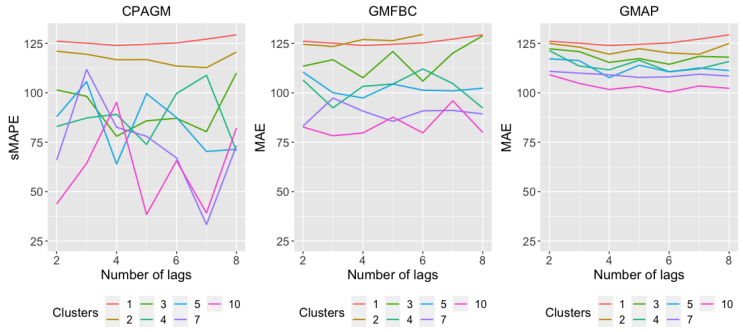

Fig. 1. Average sMAPE as a function of the number of lags in dataset M1 Yearly. Each colour corresponds to a different value of the number of clusters, K.

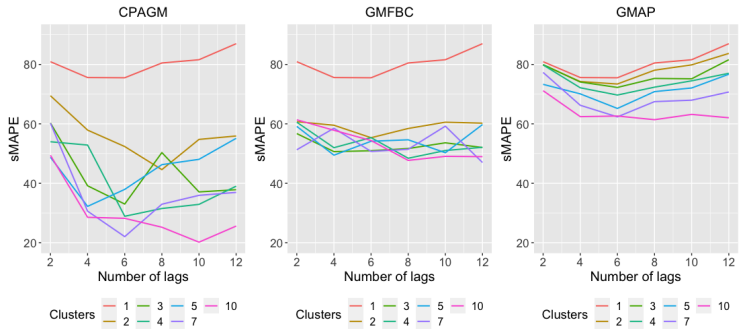

Fig. 2. Average sMAPE as a function of the number of lags in dataset M1 Quarterly. Each colour corresponds to a different value of the number of clusters, K.

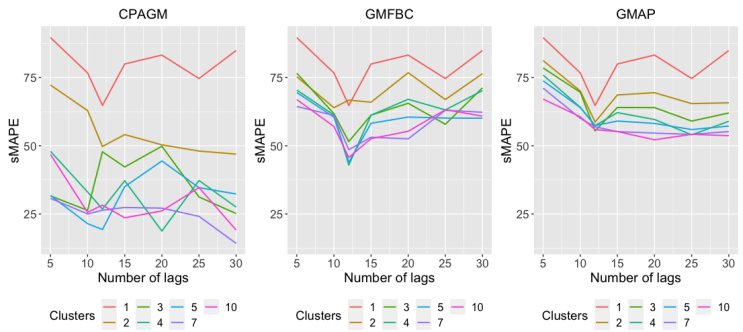

Fig. 3. Average sMAPE as a function of the number of lags in dataset M1 Monthly. Each colour corresponds to a different value of the number of clusters, K.

	LM	CPAGM ($K = 1$)	GMFBC	GMAP
M1 Yearly				
Optimal (K, l)	-	$(7, 7)$ $(l = 4)$	$(10, 3)$	$(10, 6)$
Average sMAPE	39.08	**33.34** (124.00)	78.28	100.40
M1 Quarterly				
Optimal (K, l)	-	$(10, 10)$ $(l = 6)$	$(10, 8)$	$(10, 8)$
Average sMAPE	26.27	**20.18** (75.52)	61.40	47.00
M1 Monthly				
Optimal (K, l)	-	$(7, 30)$ $(l = 12)$	$(4, 12)$	$(10, 20)$
Average sMAPE	18.51	**14.22** (64.78)	42.95	52.20

Table 5. Summary of results of Figures 1, 2 and 3. The average sMAPE obtained by the LM approach was also incorporated.

tering in additional databases from several domains (M3 and M4 competitions, medicine, finance...). In most cases, the obtained conclusions were very similar to the ones associated with dataset M1. The scores of the different methods in the considered datasets are available under request.

5 Conclusions

In this work, a clustering algorithm based on prediction accuracy of global forecasting models was introduced. The procedure is based on the traditional K-means method and relies on a two-step iterative process: (i) K global models (centroids) are fitted by considering the series belonging to each cluster and (ii) each time series is assigned to the group associated with the centroid yielding the lowest forecasting error according to the MAE metric. Although the main goal of the method is to produce a meaningful clustering partition, the nature of the iterative process makes the algorithm also an appropriate tool to be used for forecasting purposes. In fact, we expect the centroid of a given cluster to predict with high accuracy future values of the series belonging to that cluster. The proposed approach was evaluated by means of a broad simulation study where the groups were characterized by different underlying stochastic processes. The algorithm was also applied to perform clustering in classical time series datasets. Overall, the proposed technique showed an excellent performance.

References

1. Bandara, K., Bergmeir, C., Smyl, S.: Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. Expert systems with applications **140**, 112896 (2020)
2. D'Urso, P., De Giovanni, L., Massari, R.: Garch-based robust clustering of time series. Fuzzy Sets and Systems **305**, 1–28 (2016)
3. D'Urso, P., De Giovanni, L., Massari, R.: Trimmed fuzzy clustering of financial time series based on dynamic time warping. Annals of operations research **299**(1), 1379–1395 (2021)

4. D'Urso, P., De Giovanni, L., Massari, R., D'Ecclesia, R.L., Maharaj, E.A.: Cepstral-based clustering of financial time series. Expert Systems with Applications **161**, 113705 (2020)
5. D'Urso, P., Maharaj, E.A.: Autocorrelation-based fuzzy clustering of time series. Fuzzy Sets and Systems **160**(24), 3565–3589 (2009)
6. Hubert, L., Arabie, P.: Comparing partitions. Journal of classification **2**(1), 193–218 (1985)
7. Hyndman, R., Athanasopoulos, G., Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S.: Forecasting functions for time series and linear models. R package version **6** (2015)
8. Hyndman, R.J., Wang, E., Laptev, N.: Large-scale unusual time series detection. In: 2015 IEEE international conference on data mining workshop (ICDMW). pp. 1616–1619. IEEE (2015)
9. Hyndman, R.J., et al.: Another look at forecast-accuracy metrics for intermittent demand. Foresight: The International Journal of Applied Forecasting **4**(4), 43–46 (2006)
10. Liao, T.W.: Clustering of time series data: A survey. Pattern Recognit. **38**(11), 1857–1874 (2005)
11. Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E., Winkler, R.: The accuracy of extrapolation (time series) methods: Results of a forecasting competition. Journal of forecasting **1**(2), 111–153 (1982)
12. Montero-Manso, P., Hyndman, R.J.: Principles and algorithms for forecasting groups of time series: Locality and globality. International Journal of Forecasting **37**(4), 1632–1653 (2021)
13. Paparrizos, J., Gravano, L.: k-shape: Efficient and accurate clustering of time series. In: Proceedings of the 2015 ACM SIGMOD international conference on management of data. pp. 1855–1870 (2015)
14. Vilar, J.A., Alonso, A.M., Vilar, J.M.: Non-linear time series clustering based on non-parametric forecast densities. Computational Statistics & Data Analysis **54**(11), 2850–2865 (2010)