
# Time Series Clustering based on Prediction Accuracy of Global Forecasting Models

**Ángel López-Oriona**[1,2], **Pablo Montero-Manso**[3] and **José A. Vilar**[1,2]

[1]Research group MODES, University of A Coruña, Spain
[2]Center of Information and Communication Technologies (CITIC)
[3]The University of Sydney Business School

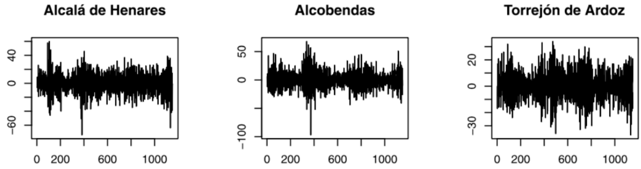
**September 23, 2022**

MODES DPTO. DE MATEMÁTICAS
UNIVERSIDADE DA CORUÑA
citic
THE UNIVERSITY OF SYDNEY

# Time Series Clustering (I)

- Time series clustering (TSC) is the problem of splitting a set of unlabelled time series into homogeneous groups so that similar series are placed together in the same group and dissimilar series are located in different groups.

- **Example**. Clustering of Spanish locations in terms of (differences of) daily concentrations of $NO_2$ (Vilar, Lafuente-Rego, & D'Urso, 2018).

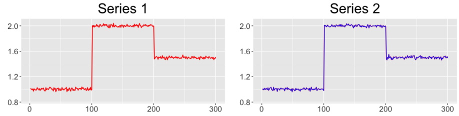



- **Dissimilarity between time series**. Several distance measures have been proposed for TSC: metrics based on spectral quantities (Caiado, Crato, & Peña, 2006), autocorrelations (D'Urso & Maharaj, 2009), geometric features (Łuczak, 2016), etc.

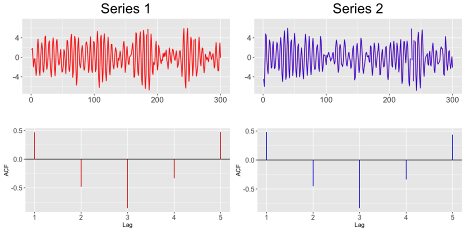
# Time Series Clustering (II)



- **Similarity based on geometric profiles**.




- **Similarity based on autocorrelations**.




- **What about similarity in terms of forecasting structures?**. Think of series of COVID-19 daily cases in two different countries sharing a common pattern. The information contained in one series is useful to predict future values of the other.

# Global Models (I)



- Usually, we are interested in the future part of each $T$-length time series $\boldsymbol{X}_t \in \mathbb{R}^T$ (vector in $\mathbb{R}^T$) up to $h$ time steps (vector in $\mathbb{R}^h$), so that we employ a forecasting function $f : \mathbb{R}^T \longrightarrow \mathbb{R}^h$.

- Let $\mathbb{X}$ be the collection of all sets of $T$-length univariate time series of finite size, i.e.,

$$\mathbb{X} = \left\{ \mathcal{X} : \mathcal{X} = \left\{ \boldsymbol{X}_t^{(1)}, \dots, \boldsymbol{X}_t^{(r)} \right\}, \text{with } r \in \mathbb{N} \text{ and } \boldsymbol{X}_t^{(i)} \in \mathbb{R}^T, i = 1, \dots, r \right\}.$$

- A global method, $\mathcal{A}_G$, is a learning algorithm taking the form

$$\mathcal{A}_G : \mathbb{X} \longrightarrow \mathbb{F}_T^h,$$

where $\mathbb{F}_T^h$ is the set of all functions with domain $\mathbb{R}^T$ and range $\mathbb{R}^h$.

- **Why using global models?**. Global models have been shown to achieve comparable forecasting accuracy or even to outperform local methods, but with far fewer parameters (Montero-Manso & Hyndman, 2021).

# Global Models (II)



- **How do we fit a global model?**
  1. Each series in $\mathcal{X}$ is lag-embedded into a matrix at a given AR order $l$.
  2. Matrices in 1 are stacked together to form one big matrix, achieving data pooling.
  3. A classical regression model (e.g., linear regression, random forest etc) is fitted to the whole matrix.

- **Toy example**. Let $T = 4$, $l = 2$ and $\mathcal{X} = \{\boldsymbol{X}_t^{(1)}, \boldsymbol{X}_t^{(2)}\}$, with $\boldsymbol{X}_t^{(1)} = (X_1^{(1)}, X_2^{(1)}, X_3^{(1)}, X_4^{(1)})$ and $\boldsymbol{X}_t^{(2)} = (X_1^{(2)}, X_2^{(2)}, X_3^{(2)}, X_4^{(2)})$. We fit a classical regression model by considering the matrix

$$\begin{pmatrix} X_1^{(1)} & X_2^{(1)} & X_3^{(1)} \\ X_2^{(1)} & X_3^{(1)} & X_4^{(1)} \\ X_1^{(2)} & X_2^{(2)} & X_3^{(2)} \\ X_2^{(2)} & X_3^{(2)} & X_4^{(2)} \end{pmatrix},$$

where the last column represents the response variable.

# A New Way of Measuring Dissimilarity



- Assume that the series $\boldsymbol{X}_t^{(i)} = (X_1^{(i)}, X_2^{(i)} \dots, X_T^{(i)})$ contains training and a validation periods of lengths $r(i)$ and $s(i)$, denoted by $\mathcal{T}^{(i)} = (t_1^i, \dots, t_{r(i)}^i)$ and $\mathcal{V}^{(i)} = (v_1^i, \dots, v_{s(i)}^i)$, respectively.

- **How to measure dissimilarity between a time series $\boldsymbol{X}_t^{(i)}$ and a global model $\mathcal{M}$?** By computing the *Mean Absolute Error* (MAE) associated with forecasting the validation period of $\boldsymbol{X}_t^{(i)}$ through the model $\mathcal{M}$.

- Distance between time series $\boldsymbol{X}_t^{(i)}$ and global model $\mathcal{M}$ is defined as

$$d_{\text{MAE}}\left(\boldsymbol{X}_t^{(i)}, \mathcal{M}\right) = \frac{1}{s(i)} \sum_{j=1}^{s(i)} \left| v_j^i - F_j^{(i)} \right|,$$

where $F_j^{(i)}$ is the prediction of $v_j^i$ by considering the global model $\mathcal{M}$.

- The larger the prediction error, the larger the distance between the series and the global model.

# Clustering based on Prediction Accuracy of Global Models (CPAGM)



- **Goal**. Given a set of $n$ time series, $\mathcal{S} = \left\{ \boldsymbol{X}_t^{(1)}, \ldots, \boldsymbol{X}_t^{(n)} \right\}$, 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 periods.
- **Iterative procedure of CPAGM algorithm**. Fix $l$ and $K$. Given a initial set of $K$ clusters $\{C_1, \ldots, C_K\}$, we fit a $l$-lagged global model, $\mathcal{M}_k$, by considering the training periods of series in $k$th cluster, $k = 1, \ldots, K$. The following steps are iterated:
  1. For $i = 1, \ldots, n$, each series $\boldsymbol{X}_t^{(i)}$ is assigned to cluster $k' = \arg\min_{k=1,\ldots,K} d_{\text{MAE}}\left(\boldsymbol{X}_t^{(i)}, \mathcal{M}_k\right)$.
  2. A new set of global models is constructed.
- **Output**. The clustering partition and the set of global models $\{\mathcal{M}_1, \ldots, \mathcal{M}_K\}$ (centroids), representing the prediction patterns.
- **Objective function**. $\sum_{k=1}^{K} \sum_{\substack{i=1: \\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} d_{\text{MAE}}(\boldsymbol{X}_t^{(i)}, \mathcal{M}_k)$, which is a sum of prediction errors.

# Evaluation of CPAGM



- **Clustering effectiveness**. Clustering quality can be assessed by means of: (i) external indexes as the Adjusted Rand Index (ARI) or (ii) internal indexes as the Xie-Beni Index (XBI). External indexes need the true partition.

- **Forecasting accuracy**. Prediction error must be assessed by considering a test set (otherwise it would be underestimated).

  1. Define the test 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 $h$ (prediction horizon). Run CPAGM method by using the set $\mathcal{S}$ as input.
  2. Given the clustering solution of Step 1, obtain global models $\{\overline{\mathcal{M}}_1, \ldots, \overline{\mathcal{M}}_K\}$considering training and validations periods.
  3. Compute the average prediction error with respect to the test set as $\frac{1}{n}\sum_{k=1}^{K}\sum_{\substack{i=1:\\ \boldsymbol{X}_t^{(i)} \in C_k}}^{n} d^*\left(\boldsymbol{X}_t^{(i)*}, \overline{\mathcal{M}}_k\right)$, where $d^*$ is any function measuring discrepancy between the actual values of $\boldsymbol{X}_t^{(i)*}$ and their predictions according to model $\overline{\mathcal{M}}_k$.

# Simulated Scenarios



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

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

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 (1). We fix $p = 12$. The vector of coefficients $\boldsymbol{\varphi}_{12} = (\varphi_1, \varphi_2, \dots, \varphi_{12})$ 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.

# Assessment Procedure



- **Alternative approaches**:
  1. Local Models (LM). A local AR model is fitted to each of the series in the collection and used to obtain the predictions. Clustering is performed by using the AR coefficients.
  2. Global models by considering Feature-Based Clustering (GMFBC) (Bandara, Bergmeir, & Smyl, 2020).
  3. Global Models by considering an Arbitrary Partition (GMAP). A random clustering partition is created and a global model is fitted within each cluster.
- $N$ time series of length $T$ were simulated from each process. The number of clusters was set to $K = 3$. The test set was constructed by considering the last $h = 8, 24$ observations of each series in Scenarios 1 and 2, respectively. Linear global models were considered. In-sample MAE was used for the reassignment step. The simulation procedure was repeated 200 times for each pair $(T, N)$.
- **Evaluation metrics**. ARI for clustering effectiveness and MAE for prediction accuracy.

# Results. Scenario 1. ARI



| $(T, N)$ | Local Models | Proposed | Bandara et al. (2020) |
|---|---|---|---|
| (20, 5) | 0.027 | **0.352**$^*$ | 0.094 |
| (20, 10) | 0.032 | **0.459**$^*$ | 0.090 |
| (20, 20) | 0.029 | **0.556**$^*$ | 0.092 |
| (20, 50) | 0.026 | **0.612**$^*$ | 0.076 |
| (50, 5) | 0.305 | **0.914**$^*$ | 0.243 |
| (50, 10) | 0.336 | **0.956**$^*$ | 0.222 |
| (50, 20) | 0.331 | **0.988**$^*$ | 0.216 |
| (50, 50) | 0.331 | **0.981**$^*$ | 0.195 |
| (100, 5) | 0.747 | **0.946**$^*$ | 0.379 |
| (100, 10) | 0.740 | **0.954**$^*$ | 0.380 |
| (100, 20) | 0.743 | **0.961**$^*$ | 0.334 |
| (100, 50) | 0.740 | **0.956**$^*$ | 0.311 |
| (200, 5) | 0.876 | **0.906** | 0.581 |
| (200, 10) | 0.854 | **0.919**$^*$ | 0.561 |
| (200, 20) | 0.820 | **0.921**$^*$ | 0.516 |
| (200, 50) | 0.800 | **0.926**$^*$ | 0.488 |
| (400, 5) | 0.897 | **0.908** | 0.719 |
| (400, 10) | 0.848 | **0.900**$^*$ | 0.725 |
| (400, 20) | 0.877 | **0.881** | 0.732 |
| (400, 50) | 0.803 | **0.872**$^*$ | 0.726 |

# Results. Scenario 1. MAE

| $(T, N)$ | Local Models | Proposed ($K = 1$) | Bandara et al. (2020) | Naive |
|---|---|---|---|---|
| (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 |

# Results. Scenario 2. ARI



| $(T, N)$ | Local Models | Proposed | Bandara et al. (2020) |
|---|---|---|---|
| $(50, 5)$ | 0.243 | **0.584**$^*$ | 0.238 |
| $(50, 10)$ | 0.259 | **0.853**$^*$ | 0.222 |
| $(50, 20)$ | 0.250 | **0.956**$^*$ | 0.219 |
| $(50, 50)$ | 0.256 | **0.980**$^*$ | 0.205 |
| $(100, 5)$ | 0.386 | **0.933**$^*$ | 0.278 |
| $(100, 10)$ | 0.387 | **0.937**$^*$ | 0.274 |
| $(100, 20)$ | 0.410 | **0.979**$^*$ | 0.277 |
| $(100, 50)$ | 0.412 | **0.986**$^*$ | 0.286 |
| $(200, 5)$ | 0.453 | **0.907**$^*$ | 0.302 |
| $(200, 10)$ | 0.478 | **0.937**$^*$ | 0.317 |
| $(200, 20)$ | 0.468 | **0.959**$^*$ | 0.306 |
| $(200, 50)$ | 0.477 | **0.972**$^*$ | 0.303 |
| $(400, 5)$ | 0.517 | **0.898**$^*$ | 0.383 |
| $(400, 10)$ | 0.510 | **0.918**$^*$ | 0.382 |
| $(400, 20)$ | 0.507 | **0.926**$^*$ | 0.368 |
| $(400, 50)$ | 0.487 | **0.921**$^*$ | 0.365 |
| $(1000, 5)$ | 0.571 | **0.846**$^*$ | 0.497 |
| $(1000, 10)$ | 0.556 | **0.841**$^*$ | 0.456 |
| $(1000, 20)$ | 0.552 | **0.867**$^*$ | 0.453 |
| $(1000, 50)$ | 0.532 | **0.877**$^*$ | 0.457 |

# Results. Scenario 2. MAE

| $(T, N)$ | Local Models | Proposed ($K = 1$) | Bandara et al. (2020) | Naive |
|---|---|---|---|---|
| $(50, 5)$ | 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 |

# Application. Clustering Datasets of M1 Competition



- We considered 3 datasets used in the M1 competition (Makridakis et al., 1982), associated with yearly (181 series), quarterly (203 series) and monthly (617 series) periodicity.
- Procedures CPAGM, GMFBC and GMAP were run for several values of $K$, namely $K \in \{1, 2, 3, 4, 5, 7, 10\}$ and $l$. We evaluated only the forecasting accuracy. We considered $h = 5$, linear global models and the in-sample MAE.
- To measure the forecasting accuracy, we considered the symmetric Mean Absolute Percentage Error (sMAPE) because some databases contain series which are recorded in very different scales. By considering the sMAPE metric, the average prediction error takes the form

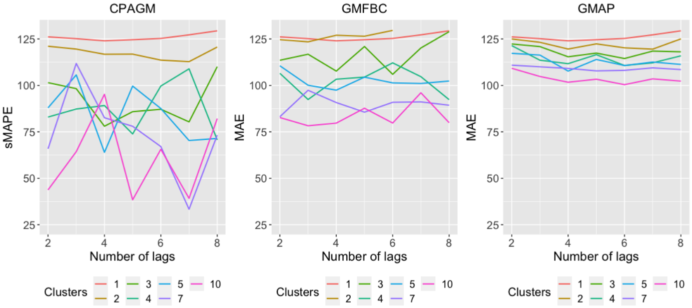
$$\frac{1}{n}\sum_{k=1}^{K}\sum_{\substack{i=1:\\ \boldsymbol{X}_t^{(i)}\in C_k}}^{n} d^*_{\text{sMAPE}}\left(\boldsymbol{X}_t^{(i)*}, \overline{\mathcal{M}}_k\right) = \frac{200}{nh}\sum_{k=1}^{K}\sum_{\substack{i=1:\\ \boldsymbol{X}_t^{(i)}\in C_k}}^{n}\sum_{j=1}^{h}\left(\frac{|X_j^{(i)*} - \overline{F}_{j,k}^{(i)*}|}{|X_j^{(i)*}|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

where $\overline{F}_{j,k}^{(i)*}$ is the prediction of $X_j^{(i)*}$ according to the global model $\overline{\mathcal{M}}_k$.

# Results. M1 Yearly



**Proposed** | **Bandara et al.** | **Naive**



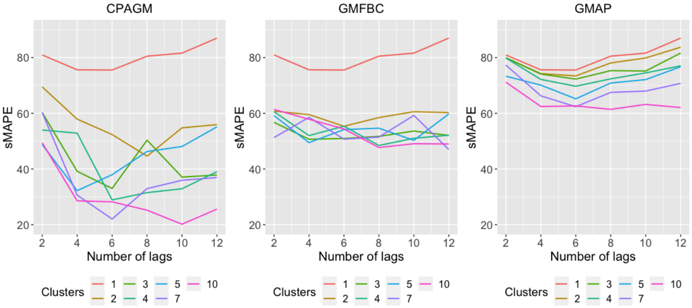

## Optimal pair and sMAPE

- **Proposed**: $(K, l) = (7, 7) \longrightarrow$ sMAPE$=$ 33.34.
- **Bandara et al.**: $(K, l) = (10, 3) \longrightarrow$ sMAPE$=$ 78.28.
- **Naive**: $(K, l) = (10, 6) \longrightarrow$ sMAPE$=$ 100.40.

# Results. M1 Quarterly



**Proposed** **Bandara et al.** **Naive**




## Optimal pair and sMAPE

- **Proposed**: $(K, l) = (10, 10) \longrightarrow$ sMAPE$= 20.18$.
- **Bandara et al.**: $(K, l) = (10, 8) \longrightarrow$ sMAPE$= 61.40$.
- **Naive**: $(K, l) = (10, 8) \longrightarrow$ sMAPE$= 47.00$.

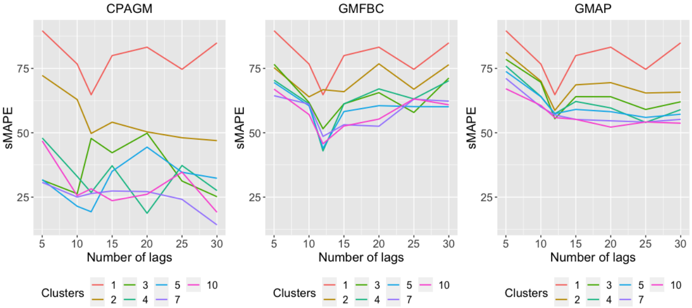
# Results. M1 Monthly

Time Series Clustering based on Prediction Accuracy of Global Forecasting Models

Ángel, Pablo and José

- Introduction
- A novel clustering algorithm
- Simulation study
- Application
- Conclusions
- References

**Proposed** **Bandara et al.** **Naive**




## Optimal pair and sMAPE

- **Proposed**: $(K, l) = (7, 30) \longrightarrow$ sMAPE$= 14.22$.
- **Bandara et al.**: $(K, l) = (4, 12) \longrightarrow$ sMAPE$= 42.95$.
- **Naive**: $(K, l) = (10, 20) \longrightarrow$ sMAPE$= 52.20$.

# Conclusions and Future Work



- We proposed a clustering algorithm based on prediction accuracy of global models, CPAGM, which produces a partition where the different clusters are associated with different prediction patterns.
- CPAGM was analysed in a simulation study containing linear processes, outperforming alternative methods in terms of both clustering effectiveness and forecasting accuracy.
- CPAGM was applied to perform clustering in datasets of M1 competition, outperforming alternative methods in terms of forecasting accuracy.
- **Future work**. Three ways to extend the current work: (i) considering simulations with nonlinear models, (ii) considering nonlinear global models (e.g., random forest) and (iii) extending the clustering algorithm to a fuzzy framework (average sum of prediction errors weighted by membership degrees).

# References



Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert systems with applications*, *140*, 112896.

Caiado, J., Crato, N., & Peña, D. (2006). A periodogram-based metric for time series classification. *Computational Statistics & Data Analysis*, *50*(10), 2668–2684.

D'Urso, P., & Maharaj, E. A. (2009). Autocorrelation-based fuzzy clustering of time series. *Fuzzy Sets and Systems*, *160*(24), 3565–3589.

Łuczak, M. (2016). Hierarchical clustering of time series data with parametric derivative dynamic time warping. *Expert Systems with Applications*, *62*, 116–130.

Makridakis, S., Andersen, A., Carbone, R., Fildes, R., Hibon, M., Lewandowski, R., ... Winkler, R. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of forecasting*, *1*(2), 111–153.

Montero-Manso, P., & Hyndman, R. J. (2021). Principles and algorithms for forecasting groups of time series: Locality and globality. *International Journal of Forecasting*, *37*(4), 1632–1653.

Vilar, J. A., Lafuente-Rego, B., & D'Urso, P. (2018). Quantile autocovariances: a powerful tool for hard and soft partitional clustering of time series. *Fuzzy Sets and Systems*, *340*, 38–72.

# Thank you for your attention!