HACKATHON ON MLGrid 36

Joana Laim
joana.laim@ua.pt
Simão Machado
simaomachado@ua.pt

Mathematics Department
University of Aveiro
Aveiro, Portugal

Abstract

Clustering time series is often hindered by high dimensionality and noise. We propose an unsupervised framework that compresses each series into four interpretable features describing the decay rate and sign pattern of its autocorrelation (ACF) and partial autocorrelation (PACF). Using these features, we cluster simulated ARMA signals with K-Means, Ward's method, and DBSCAN, evaluating quality via silhouette, Calinski-Harabasz, Davies–Bouldin, ARI, and NMI. The engineered features yield clear separation: k-means achieves the best internal scores and clusters tend to group series generated by similar ARMA (p,q) settings, preserving interpretability.

By contrast, clustering a 95 %-variance PCA space results in poorer, less stable partitions, highlighting the benefit of domain-guided features. Overall, a minimal autocorrelation-based representation can power effective, explainable time-series clustering and expose latent dynamical structure.

1 Introduction

Unsupervised learning, particularly clustering analysis, is a valuable approach for uncovering latent structure in unlabeled data. In this work, we aim to cluster time series based on feature representations derived from their autocorrelation structure, with the goal of identifying natural groupings in the data and understanding the characteristics that distinguish each group.

To achieve this, we apply and compare several clustering algorithms, including k-means, DBSCAN, and agglomerative hierarchical clustering. Rather than working with the raw time series directly, we transform them into a reduced set of interpretable features that summarize key autocorrelation behaviors. Although the dataset includes true class labels defined by parameter combinations, these are used strictly for performance evaluation, allowing the clustering process to remain entirely unsupervised.

This approach allows us to assess which clustering strategies best capture the underlying structure of the data and whether the extracted features preserve enough information to potentially support a predictive model in future work.

2 State of The Art

Recent work has explored clustering time series by leveraging their underlying autoregressive structure. Hoare et al. (2022) propose K-ARMA, a model-based method that integrates AR/ARMA fitting into the k-means framework to improve robustness and convergence [1]. Ren and Barnett (2022) introduce autoregressive mixture models that cluster based on autocovariance matrices and Wishart mixtures [2]. Similarly, Ye et al. (2020) use kernel-based ARMA modeling for clustering time series on non-Euclidean spaces [3]. While these approaches rely on explicit time series modeling, they motivate our use of autocorrelation-derived features (such as ACF and PACF decay rates) as compact and interpretable representations for clustering.

3 Exploratory Data Analysis

3.1 Dataset Description

The dataset comprises 18,000 time series samples, uniformly distributed across 36 classes. Each class corresponds to a unique pair (p,q) with $p,q \in \{0,\ldots,5\}$, resulting in 500 instances per class. Each sample is described by 88 features: 41 values from the autocorrelation function (ACF), 41 from the partial autocorrelation function (PACF), and several auxiliary parameters such as p, q, w, j, and T.

3.2 Data Preprocessing

We started by checking for missing values, and none were found. From the original 88-feature dataset, only the 82 numerical features corresponding to the ACF and PACF sequences were retained for unsupervised analysis, and the class labels (p,q) were stored separately for later evaluation.

To better understand the relationships between features, a Pearson correlation matrix was computed. The features `ACF_0` and `PACF_0` appeared blank in the matrix, as they take the constant value 1 across all observations. Since correlation is undefined for variables with zero variance, these columns were removed from the dataset. Their lack of variability also makes them uninformative for clustering tasks.

3.3 Initial Feature Analysis

The initial grouping with the entire set of 80 dimensions features was performed using k-means, supported by cluster diagnosis tools such as the elbow method, silhouette, and Calinski-Harabasz scores. To interpret patterns, the ACF and PACF distributions were plotted separately for each cluster using boxplots, revealing consistent trends in decay and early-lag sign structure, as illustrated in Figure 1. These observations indicated a high degree of redundancy within each feature set and motivated the construction of a compact feature representation using decay rates and sign-pattern categories.

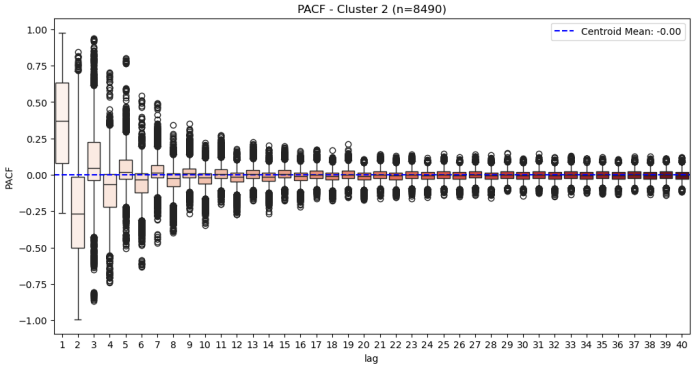

Figure 1: PACF distribution for Cluster 2, illustrating decay and sign trends across lags.

3.4 Dimensionality Reduction

To address the issue observed in the original 80 features, we extracted four compact and interpretable features from the ACF and PACF sequences. Specifically, we modeled the absolute values of each sequence using an exponential decay function of the form $y(h) = e^{-bh}$, where h is the lag and b is the decay rate. This parameter b captures the rate at which autocorrelation decreases over time and serves as a summary of the temporal memory in the sequence.

In addition, we computed a categorical sign-pattern feature to describe the general shape of the early lags: $+1$ if the values are positive, -1 if negative, and 0 if alternating. These decay and sign features were computed separately for ACF and PACF, resulting in a total of four features per instance. This low-dimensional representation retains key autocorrelation behavior while reducing noise and facilitating more robust clustering.

Before clustering, the decay features were standardized using z-score normalization, and the categorical sign-pattern features were encoded using one-hot encoding. This ensured that all features contributed equally to distance-based clustering algorithms.

In parallel, a PCA was also applied to the original ACF/PACF features for comparative purposes. This approach provided an alternative representation for clustering and is discussed in Section 5.

4 Methodology

Two feature spaces were evaluated:

(i) a four–dimensional, interpretable set derived from ACF/PACF decay rates and sign patterns.

(ii) a PCA projection retaining 95 % variance.

Three clustering algorithms were applied to each space: k-means with a smart initialization process (k-means ++), k=5 and k=4 for *(i)* and *(ii)*, respectively, Ward agglomerative clustering (cut at k=5 for *(i)* and k=4 for *(ii)*), and DBSCAN (ε=0.44, `minPts`=9) and (ε=1.3, `minPts`=23) for *(i)* and *(ii)*, respectively.

The clustering performance was evaluated using standard internal validation metrics, namely, silhouette coefficient, Calinski–Harabasz (CH), and Davies–Bouldin index (DBI), and external criteria, in particular, adjusted Rand index (ARI), normalized mutual information (NMI). The later compares cluster assignments to the known class labels.

5 Results and Discussion

5.1 Clustering performance

Table 1: Evaluation of the three clustering algorithms on *(i)*.

Algorithm	k	Sil.	DBI	CH	ARI	NMI
k-means	5	**0.329**	**1.065**	**6 710.8**	**0.010**	0.047
DBSCAN	5	0.274	1.591	2 324.7	0.005	0.031
Hier-Ward	5	0.302	1.119	5 346.5	0.009	**0.049**

Table 1 summarizes the outcomes of *(i)*. K-Means achieved the best overall performance, with the highest silhouette score and CH index and lowest DBI, indicating relatively well-separated clusters and closely related objects in a cluster. Its external metrics were also slightly higher than those of DBSCAN and Ward's method, though all external scores are low due to the cluster count of 5 being far fewer than the 36 true classes. The hierarchical Ward clustering showed intermediate performance, while DBSCAN struggled slightly more under these feature conditions. It formed five clusters and marked $0,60\%$ of points as outliers (noise), achieving a lower silhouette score and the smallest CH value.

Overall, K-Means provided more cohesive clustering in this feature space, capturing structure that the other methods missed.

Table 2: Internal and external metrics obtained on *(ii)*.

Algorithm	k	Sil.	DBI	CH	ARI	NMI
k-means	4	**0.313**	**1.066**	**1 612.7**	**0.006**	**0.047**
DBSCAN	1	NaN	NaN	NaN	0.000	0.00
Hier-Ward	4	0.288	1.104	1 305.1	0.005	**0.047**

Table 2 shows that transferring the same three algorithms to the 95 %-variance PCA space leads to noticeably weaker or less stable partitions. Although K-Means still attains the best silhouette and CH within this space, both values are far below their counterparts in the handcrafted space (Table 1). Hierarchical–Ward follows the same trend, and DBSCAN only forms a single dense cluster, yielding no meaningful internal scores. The external indices (ARI and NMI) also remain at or below the values obtained with the four interpretable features.

Beyond the numbers, the principal components lack physical meaning, hence, the handcrafted feature space is preferred for both performance and interpretability.

5.2 Cluster Characteristics

The clusters obtained from K-Means were interpretable in terms of the original (p,q) classes and autocorrelation behaviors. In other words, series generated by similar ARMA configurations tended to fall into the same cluster. Furthermore, the four-feature centroids of the K-Means clusters exhibit clear differences that align with domain intuition. For example, some clusters have a significantly higher ACF/PACF decay rate (indicating shorter memory) while others decay more slowly, and the sign-pattern features differ. These patterns suggest that our engineered features effectively captured the underlying signal dynamics, making the cluster groupings meaningful and explainable.

Table 3: Centroids of the five K-Means clusters on *(i)*. *Sign* columns report the expected sign, computed as ACF_sign_mean $= (-1)\,p_{-1} + 0\,p_0 + 1\,p_{+1}$ and analogously for PACF, where p_{-1}, p_0, p_{+1} are the one hot centroid values.

Cluster	**Decay**		**Sign (mean)**	
	ACF	**PACF**	**ACF**	**PACF**
0	0.190	-1.600	0.150	-0.370
1	0.490	0.420	0.000	0.000
2	0.130	0.300	1.000	0.150
3	0.340	0.270	-0.120	-1.000
4	-2.150	0.070	0.310	-0.210

Table 3 illustrates this, showing distinct centroid profiles per cluster corresponding to different autocorrelation decay behaviors and sign structures. For example, cluster 1 shows the fastest ACF and PACF decay rates, meaning a rapid drop-off that indicates very short memory, consistent with low-order ARMA models (small p and q). The near-zero sign features corroborate this "almost white-noise" behaviour. On the other hand, cluster 4 has the slowest ACF decay together with a small positive PACF decay. The extended ACF tail that diminishes gradually suggests high persistence, that is, this cluster gathers the more strongly correlated series, including many with low MA order but higher AR contribution.

Our interpretation is validated when plotting a heatmap of cluster membership by true class. K-Means cluster predominantly groups a distinct combination of (p,q) values (i.e. specific ARMA orders), rather than a random mix of all 36 classes.

6 Conclusion

In this study, we demonstrated that a small set of interpretable autocorrelation features can successfully drive the unsupervised clustering of time series. Synthesizing each series down to just four descriptors (ACF/PACF decay rates and sign patterns) proved sufficient to form clusters that reflect distinct autocorrelation characteristics of the series. Among the clustering algorithms tested, K-Means produced the most coherent and relatively well-separated clusters, as evidenced by its superior silhouette, CH, and DBI scores and slightly higher agreement with true (p,q) classes. This suggests that the chosen features preserve the critical information needed to discern latent groupings in the data. The clusters were also interpretable in a way each cluster corresponded to a particular profile of autocorrelation decay speed and initial sign, linking the unsupervised results back to time-series properties.

For future work, we plan to explore a broader set of techniques, including ensemble clustering, to test the robustness of our approach. We also intend to apply our four autocorrelation features in supervised learning (classification) and to validate the method on real-world time-series data. These steps will help establish the method's generalization and practical utility.

References

[1] Derek O. Hoare, David S. Matteson, and Martin T. Wells. K-arma models for clustering time series data, 2022. URL `https://arxiv.org/abs/2207.00039`.

[2] Benny Ren and Ian Barnett. Autoregressive mixture models for serial correlation clustering of time series data, 2022. URL `https://arxiv.org/abs/2006.16539`.

[3] Cong Ye, Konstantinos Slavakis, Pratik V. Patil, Johan Nakuci, Sarah F. Muldoon, and John Medaglia. Network clustering via kernel-arma modeling and the grassmannian the brain-network case, 2020. URL `https://arxiv.org/abs/2002.09943`.