24th International Conference & Exhibition on Electricity Distribution (CIRED)

12-15 June 2017

Session 5: Planning of power distribution systems

Comparing time-series clustering approaches for individual electrical load patterns

ISSN 2515-0855
doi: 10.1049/oap-cired.2017.1222
www.ietdl.org

Zacharie De Greve[1] ✉, Fabian Lecron[2], François Vallee[1], Gerard Mor[3], Daniel Perez[3], Stoyan Danov[3], Jordi Cipriano[3]

[1]Power Electrical Engineering Unit, University of Mons, Mons, Belgium
[2]Eng. Innovation Management, University of Mons, Mons, Belgium
[3]International Centre For Numerical Methods in Engineering (CIMNE), Universitat Politécnica de Catalunya (UPC), Terrassa, Spain
✉ *E-mail: zacharie.degreve@umons.ac.be*

Abstract: This work positions the task of grouping electricity load time series among the vast field of clustering, and highlights corresponding research issues. A selection of the most performant time-series clustering approaches from the signal processing community are compared on the same dataset, composed by domestic electricity load profiles from Spain. The cross-correlation-based distance of Paparrizos and Gravano (2015) is shown to provide the best trade-off between clustering accuracy and CPU times.

1 Introduction

Electricity distribution networks have not been historically designed for hosting decentralised generation. This can give rise to technical problems such as over-voltages and/or congestions, which significantly affect the operation and planning tasks and are even reinforced by the stochastic nature of renewable production and electrical load. To face these challenges, the community is proposing tailored network management strategies, founded on advanced optimisation algorithms, which are mainly based on the following (simple) rule: use energy where and when it is produced if possible. Practical implication of this rule consists for instance in the implementation of demand response strategies [1], in the installation of storage devices [2], and/or in the setting up of microgrids, which target a quasi-total energy autonomy from the main grid [3].

The performance of these strategies is strongly influenced by the accuracy of data models, which aim at representing as best as possible the state of the studied system (i.e. the electricity distribution network) given the uncertain nature of renewable, load and market quantities. More generally, the term ‘system’ may also encompass transmission networks or production portfolio managed by energy producers for instance, which face the similar needs in terms of observability, even if the objectives are different.

This paper focuses more particularly on the case of electrical load profiles. Over the past years, an increasing interest in applying machine learning techniques to such data has been shown by the actors involved in electricity distribution systems. Prediction and clustering are for instance two tasks which attract a lot of attention.

On the one hand, accurate predictions at a given time horizon obviously bring valuable information to the distribution system operator (DSO) for managing his network (e.g. for quantifying the amount of load flexibility he will need to activate in day ahead). Clustering techniques aim, on the other hand, at finding the structure which is inherent to the data, by merging similar patterns into groups (or clusters). It may pertain to the grouping of different clients (e.g. for customer segmentation in market applications), and/or to the generation of representative patterns for a given client (for instance typical daily profiles useful in distribution system planning).

The literature is very mature concerning clustering techniques, but is mainly focused on traditional machine learning applications driven by applications such as audio/video processing, biology etc. Dressing a state of the art in general clustering would be out of the scope of the present paper (Jain [4] provides a good general overview of the field). The trend is however evolving, so that recent references are proposing approaches for grouping electrical load patterns [5] or wind production time series [6] for instance.

Nevertheless, when it comes to time-series clustering, namely to the grouping of temporal data patterns such as individual electricity load profiles, a particular attention must be paid on the distance employed to compare the time series. A meaningful example is obtained by considering the fact that two time series which are similar in shape but different in phase (or in alignment) may belong to the same cluster. Indeed, a customer will not switch on his washing machine at exactly the same time every day. Distances that are satisfying this property are named shift-invariant distances. In that regard, the distances employed classically in audio/video processing (e.g. Euclidian, Mahalanobis etc.) and in most of the literature pertaining to electrical systems, may not be appropriate.

The signal processing community has proposed various shift invariant distances, among which dynamic time warping or DTW [7] is a popular one. Its computation is however demanding in CPU time, preventing it to be employed in a Big Data context. Very recently, a cross-correlation-based (CCB) distance, which is not suffering from this drawback, has been proposed in [8]. A performance analysis of these distances on electrical quantities such as load patterns is however missing in the literature, to the best of the authors’ knowledge.

In that context, the objectives of the present paper are threefold, and consist in:

(i) Positioning the task of electricity load grouping among the general field of clustering, by highlighting the peculiar characteristics of load data (see Section 2).
(ii) Applying the most recent time-series clustering techniques from the signal processing community (e.g. [8]) to electrical load patterns, and compare the results with more traditional approaches such as *k*-means/*k*-medoids [9] (see Section 3).
(iii) Comparing these methods on the same database of domestic electricity load patterns, obtained from the European EMPOWERING project [10].

2 Classification of time-series clustering algorithms

2.1 Feature- and model-based approaches

In feature-based approaches, an appropriate transformation is first applied on the raw data in order to extract meaningful and non-redundant information (or features). The clustering algorithm itself is then applied on the transformed data. By doing so, it is expected that the complexity of the chosen clustering algorithm will be decreased, while yielding similar or better performances. Most of the algorithms can be employed on features and on raw data as well, so that they will be described below in Section 2.4. In [5] for instance, a discrete wavelet transform (or DWT) is first computed on raw electrical load data, before performing a g-means (a generalisation of the k-means) algorithm for grouping the obtained patterns.

Model-based techniques rely on the other hand on the assumption that the data comes from a specific parametric structure. A Gaussian mixture distribution is often adopted in machine learning applications, but in the case of time-series clustering, one can for instance assume that the studied series follow autoregressive integrated moving average (or ARIMA) models (see, e.g. [11]). In [12], cepstral coefficients, commonly used in speech processing, are employed to discriminate ARIMA time series.

The main characteristic of feature- and model-based approaches is that they rely on context-dependent information, and require an a priori knowledge on the studied data. In that way, the best feature transformation or the best parametric model may change depending on the application, which is a drawback in our case: the electrical load profiles can correspond to consumers who have significantly different behaviours and habits. Such techniques are not illustrated in this paper, since general approaches which require as less user expertise as possible are targeted.

Note that the families of spectral and kernel clustering techniques, a recent review of which can be found in [13], belong to the feature-based category and are able to provide non-linear separation hyperplanes between clusters, without any particular assumption on the data. In spectral methods, clustering is performed on a (reduced) set of eigenvalues computed on the similarity matrix (in the sense of graph theory) of the data. On the other hand, kernel techniques intend to map the initial data into a higher dimensional space, in which a linear partition is computed, resulting in a non-linear partition in the input space. Support vector clustering (SVC) [14] is an example of kernel clustering. Such methods gave promising results in various applications, but the computational cost is often very high, preventing their use in real-life applications. For that reason, they will not be studied in the present paper.

2.2 Raw-based approaches

The clustering algorithm is directly applied on raw data in this case (or on data which has undergone minor transformations such as normalisation, see below). The complexity of the feature-extraction process, which is context dependent by nature, is thus avoided, but is reported mainly in the distance employed to compare objects.

2.2.1 Distances for comparing time series

When it comes to time-series clustering, particular attention must be paid on the distance employed to compare the sequences. In general, the chosen distance may have to satisfy several invariances which depend on the application, according to the classification initiated in [8, 15];

- Scaling and translation invariances. A distance which is invariant to scaling and translation is able to recognise the similarity between time serieswhich have different scales (or amplitudes), and/or which are translated along the y-axis (offset). Usually, a normalisation step (i.e. substracting the mean and dividing by the standard deviation) is applied on the raw data to ensure such an invariance.
- Shift (or phase) invariance. Two time series which are similar in shape but different in phase (or in global alignment) may belong to the same cluster. It is also true for a local alignment, i.e. if some portions are aligned and others not. For example, in the case of electrical load patterns, a customer will not switch on his washing machine at exactly the same time every day, which results in similar time series with a possible different local alignment.
- Sampling invariance. This is required for comparing sequences which have been recorded with different sampling frequencies and have therefore different lengths, by either stretching the shortest one or shrinking the longest one.
- Occlusion invariance. Recorded time series may have missing data due to sensor malfunction, loss of the transmission channel, data storage equipment failure etc. Occlusion-invariant distances are robust to holes in the series which are compared.
- Complexity invariance. Recording the same signal in a low- and high-noise environment generates series of different complexities, which might result in a classification in different clusters with classical distances.

Comparing electric load patterns from a given geographical area mainly require to satisfy the phase invariance as explained above. The problematic of scale and translation invariance, if needed, can be tackled by normalisation. Sampling invariance is not required here, since the electrical load vectors are recorded at the same frequency. Moreover, using complexity invariant distances is not relevant in the present paper, given the test database characteristics (see Section 3 for more details). Batista *et al.* [15] propose however to correct any distance by a complexity factor computed for each pair of objects in order to satisfy such an invariance. Occlusion invariance, on the other hand, may be needed since missing data is quite common in real-life databases. Two strategies can be adopted to handle that: one can employ distances able to ignore the holes when comparing objects, or rather fix the holes in pre-processing using straightforward or more advanced interpolation techniques, depending on the size of the holes.

In this work, the focus is mainly on shift invariance, given the peculiar characteristics of nodal load data. All the distances available in the literature will therefore not be presented (a good review of time-series distances can be found in [16] for instance), but the most relevant to the best of authors' knowledge are selected and presented below.

In that context, according to [17], the classical Euclidean distance is a good first candidate since it provides acceptable performance. It is however significantly outperformed by the DTW distance [7], which can be seen as an extension of the Euclidean allowing for a local alignment of subsequences, at the expense of higher computational costs: the distance is indeed computed with a quadratic complexity $O(n^2)$, with n the size of the sequences (compared to the linear complexity of the Euclidean distance). Several improvements have been proposed in the literature in order to speed-up the DTW process. These are mainly based on the use of lower [18, 19] and upper bounds [20] of the DTW distance, which can be extracted in linear time. More recently, a distance based on the cross-correlation between signals has been proposed in [8]. An interpretation in terms of discrete Fourier transforms permits to reach a $O(n \log n)$ complexity, provided that a fast Fourier transform algorithm is employed.

2.2.2 Clustering algorithms

Clustering algorithms are mainly divided into two classes, namely hierarchical and partitional. Hierarchical algorithms [21] aim at finding a recursive structure in the data (the results of the clustering procedure are often presented under the form of a tree or dendrogram). They have a time and memory complexity of $O(N^2)$, with N the number of sequences to compare, and are therefore used on relatively small databases [22]. Partitional algorithms do not provide a hierarchy among clusters, but try to

CIRED, Open Access Proc. J., 2017, Vol. 2017, Iss. 1, pp. 2165–2168

compute separating planes between them, which can be linear (as in *k*-means) or not, as explained above. *K*-means and *k*-medoids (which differ on the way the cluster centroids are computed, see Section 2.5) are well-known examples. The class of density-based methods belongs to partitional clustering. Clusters are built based on their density, which allow for the identification of outliers and of clusters of arbitrary shape, compared to *k*-means which tends to discover spherical clusters (see [23]).

2.2.3 Representative object computation

Computing a representative object (or prototype) for each cluster is an important task. Indeed, such objects are expected to gather the most important characteristics of the similar series which are present in the same cluster. In our case, the prototypes can be directly associated with typical days of electrical consumption for a given client, or even to typical consumption profiles among a set of clients, as explained in Section 1. A common approach consists in taking the arithmetic mean in the Euclidean distance sense – or centroid – of the cluster objects as a prototype. In the case of time series however, particular attention must be paid on the distance employed to extract the prototypes. For instance, Euclidean averaging of misaligned series can lead to the apparition of spurious (or non-physical) modes in the prototype pattern, which must be avoided. Again, the literature is abundant on that topic, but selected approaches are presented here. The partition around medoid (PAM) algorithm selects for instance, among the cluster objects, the one which minimises the average distance to all the others (this corresponds to the well-known *K*-medoids algorithm). Dynamic barycenter averaging (DBA) appears however to be the most efficient averaging technique when DTW distance is employed [24]. The shape-extraction technique of [8] is preferred when using the CCB distance.

3 Experiments

3.1 Database

The EMPOWERING project [10] has gathered hourly profiles of more than 8000 clients in the area of Barcelona, Spain, for approximately 2 years, in a database of 3.3 Gb. Tests are conducted here on a subset of three clients, in order to illustrate various clustering algorithms, even the slowest ones, without facing computer memory limits. The clustering of similar profiles is based on daily load curves, accounting 365 daily curves per year per user.

3.2 Methodology

As explained, the focus is on the shift invariance in this work. Data is normalised in pre-processing, and sequences with missing data are simply removed from the database. Different distances with their best prototyping (or representative object computation) method according to the literature are compared on the same database, namely:

Euclidean: the Euclidean distance with a PAM algorithm;
DTW: the DTW distance of [7] with DBA;
DTW_LB: an accelerated version of DTW distance using lower bounds as in [19] along with DBA;
DTW_LUB: a recent version of DTW using lower and upper bounds with a density-based prototype selection [20]
CCB: the CCB distance of [8] with its shape-extraction procedure.

The same partitional clustering algorithm, extracted from the *k*-means/medoids paradigm, is chosen for comparison purposes (except for the approach *DTW _LUB*, which is density based). The most common cluster internal evaluation metrics (i.e. which do not require the correct partition to be known a priori) are computed [25], but the results are reported for the Silhouette index only for the sake of conciseness. Methodologies are implemented in the open-source R language [26], using mainly the 'dtwclust' package [22].

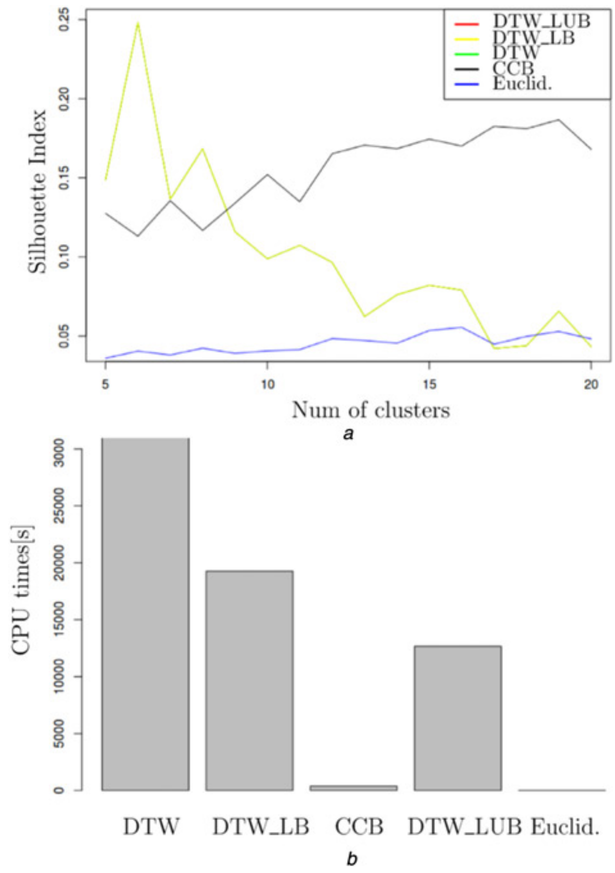

Fig. 1 *Results of the clustering experiments*
(a) Silhouette index as a function of the number of clusters, ***(b)*** CPU times for DTW, DTW_LB, DTW_LUB, Euclidean and CCB algorithms

4 Results

Fig. 1*a* depicts the Silhouete index (to be maximised) for a number of clusters varying from 5 to 20, for the five methods. The superiority of shift-invariant distances can be observed. It is verified that DTW-based distances provide the same indices (i.e. green, yellow and red lines are merged), which means that using lower and upper bounds instead of the exact distances does not degrade the results. CPU times (excluding the indices computation which can be time demanding) are reported in Fig. 1*b*: it appears that the methodology which provide the best compromise between accuracy and computational cost is the CCB.

5 Conclusion and perspectives

This paper has demonstrated the superiority of the CCB distance of [8] over a selection of the most performant time-series clustering approaches [7, 19, 20] for grouping electrical load profiles. This provides the best trade-off between accuracy (in terms of internal clustering evaluation indexes according to the definitions of [25]), and CPU times. A next step would be to include the CCB distance in more robust partitioning algorithms such as spectral, kernel and density-based methods.

This work consists in a first study: the comparison of clustering techniques on the same dataset should be extended to other techniques for being exhaustive, and the satisfaction of other types of distance invariances, such as occlusion and complexity more particularly, should be investigated. The present work intends to provide a comprehensive introduction to the vast field of clustering, with an emphasis on the issues to consider when it comes to load data.

The CPU time argument is even more relevant if one consider the data tsunami which is expected from the electricity distribution networks. In that Big Data context, fast distances in combination with multi-step parallelizable clustering approaches such as in [5] should be favoured.

6 References

1 Wang, Z., Paranjape, R.: 'Optimal residential demand response for multiple heterogeneous homes with real-time price prediction in a multiagent framework', *IEEE Trans. Smart Grids*, 2017, **8**, (3), pp. 1173–1184

2 Sossan, F., Torregrossa, D., Namor, E., *et al.*: 'Control of a battery energy storage system accounting for the charge redistribution effect to dispatch the operation of a medium voltage feeder'. Proc. of the IEEE PES PowerTech Conf., Eindhoven, 2015, p. 6

3 Stevanoni, C., Vallée, F., De Grève, Z., *et al.*: 'On the use of game theory to study the planning and profitability of industrial microgrids connected to the distribution network'. Proc. of the 24th CIRED Conf., same issue, 2017

4 Jain, A.K.: 'Data clustering: 50 years beyond *k*-means', *Pattern Recognit. Lett.*, 2010, **31**, (8), pp. 651–666

5 Mets, K., Depuydt, F., Develder, C.: 'Two-stage load pattern clustering using fast wavelet transformation', *IEEE Trans. Smart Grids*, 2016, **7**, (5), pp. 2250–2259

6 Picart, B., Gosselin, B., De Grève, Z., *et al.*: 'A fast clustering approach for generating wind profiles in electrical power system studies', 2016, submitted for publication

7 Sakoe, H., Chiba, S.: 'Dynamic programming algorithm optimization for spoken word recognition', *IEEE Trans. Acoust. Speech Signal Process.*, 1978, **26**, (1), pp. 43–49

8 Paparrizos, J., Gravano, L.: 'K-shape: efficient and accurate clustering of time series'. Proc. of the 2015 ACM SIGMOD Int. Conf. on Management of Data, 2015, pp. 1855–1870

9 Bishop, C.M.: 'Pattern recognition and machine learning' (Springer, 2006)

10 Empowering project, Available at http://iee-empowering.eu/en/, 2017

11 Maharaj, E.A.: 'Clusters of time series', *J. Classif.*, 2000, **17**, (2), p. 297314

12 Kalpakis, K., Gada, D., Puttagunta, V.: 'Distance measures for effective clustering of ARIMA time-series', in Cercone, N., Lin, T.Y., Wu, X., (Ed.): 'Proc. 2001 IEEE Int. Conf. on Data Mining', 2001, p. 273280

13 Filippone, M., Camastra, F., Masulli, F., *et al.*: 'A survey of kernel and spectral methods for clustering', *Pattern Recognit.*, 2007, **41**, pp. 176–190

14 Ben-Hur, A., Horn, D., Siegelmann, H.T., *et al.*: 'Support vector clustering', *J. Mach. Learn. Res.*, 2001, **2**, pp. 125–137

15 Batista, G.E., Keogh, E.J., Tataw, O.M., *et al.*: 'CID: an efficient complexity-invariant distance for time series', *Data Min. Knowl. Discov.*, 2014, **28**, (3), pp. 634–669

16 Montero, P., Vilar, J.A.: 'TSclust: an R package for time series clustering', *J. Stat. Softw.*, 2014, **62**, (1), pp. 1–43

17 Wang, X., Mueen, A., Ding, H., *et al.*: 'Experimental comparison of representation methods and distance measures for time series data', *Data Min. Knowl. Discov.*, 2013, **26**, (2), p. 275309

18 Keogh E, E., Ratanamahatana, C.A.: 'Exact Indexing of dynamic time warping', *Knowl. Inf. Syst.*, 2005, **7**, (3), pp. 358–386

19 Lemire, D.: 'Faster retrieval with a two-pass dynamic-time-warping lower bound', *Pattern Recognit.*, 2009, **42**, (9), pp. 2169–2180

20 Begum, N., Ulanova, L., Wang, J., *et al.*: 'Accelerating dynamic time warping clustering with a novel admissible pruning strategy'. Proc. of the ACM SIGKDD Conf. on Knowledge Discovery and Data Mining, 2015, 10 pages

21 Kaufman, L., Rousseeuw, P. J.: 'Finding groups in data: an introduction to cluster analysis' (John Wiley & Sons, 2009), p. 344

22 Sardà-Espinosa, A.: 'dtwclust: time series clustering along with optimizations for the dynamic time warping distance', R package version 2.2.1., 2016, Available at https://CRAN.R-project.org/package=dtwclust

23 Ware, V.S., Bharathi, H.N.: 'Study of density based algorithms', *Int. J. Comput. Appl.*, 2013, **69**, (26)

24 Petitjean, F., Ketterlin, A., Ganarski, P.: 'A global averaging method for dynamic time warping, with applications to clustering', *Pattern Recognit.*, 2011, **44**, (3), p. 678693

25 Arbelaitz, O., Gurrutxaga, I., Muguerza, J., *et al.*: 'An extensive comparative study of cluster validity indices', *Pattern Recognit.*, 2013, **46**, (1), pp. 243–256

26 R project, Available at http://www.r-project.org, 2017