EPiC Series in Engineering

Volume 3, 2018, Pages 1123–1130

HIC 2018. 13th International Conference on Hydroinformatics

Categorisation of urban water consumptions

Joaquim Leitão[1], Nuno Simões[2], José Alfeu Marques[2], Paulo Gil[3], Bernardete Ribeiro[1], Alberto Cardoso[1]

[1] Centre for Informatics and Systems of the University of Coimbra (CISUC), University of Coimbra, Portugal
jpleitao@dei.uc.pt, bribeiro@dei.uc.pt, alberto@dei.uc.pt

[2] Marine and Environmental Sciences Centre of the University of Coimbra (MARE UC), University of Coimbra, Portugal
nunocs@dec.uc.pt, jasm@dec.uc.pt

[3] Centre of Technology and Systems (UNINOVA-CTS), Universidade NOVA de Lisboa, Portugal
psg@fct.unl.pt

Abstract

Achieving an optimised management of water supply infrastructures is a very important and challenging task, namely in urban environments. The identification and prediction of actual water consumption patterns can be exploited to improve the overall performance of water supply infrastructures. This work considers the application of pattern recognition techniques on water consumption time-series to quantify the time distribution of common consumption behaviours in urban environments. Three groups representing typical consumption patterns have been considered: one characterised by residual consumptions, which occur during the summer months of June and July, while the remaining two consist of significant consumption during the day, with differences taking place during night periods – the first group, more prevalent during warmer months, is represented by higher consumptions during the night, when compared with the second group, more representative of colder months, but showing also some expression all year round. Results also demonstrate that an automatic categorisation of urban water consumptions can be carried out along with the identification of specific time periods in which each pattern occurs.

Keywords: Pattern recognition, time-series clustering, water consumption patterns, water management.

1 Introduction

Water is the most important natural resource in our planet and a key element to the settlement and growth of communities. Water demands are subjected to changes over time, conditioned by factors such

as climate and geographical features, social and economic conditions, buildings scattering and population density, just to name out a few.

A proper understanding of water consumption patterns in urban environments is extremely important to water supply companies. By characterising consumption patterns, excess water volumes retained in reservoirs can be adjusted with both environmental, economic and energetic impacts. The current work presents a contribution in water supply management by analysing historic urban water consumptions, identifying common consumption behaviours.

The remainder of this document is organised as follows: Section 2 introduces the problem and the used data set. Section 3 discusses the adopted methodology, while Section 4 presents computed consumption categories. Finally, Section 5 concludes the document.

2 Problem and Data Set Description

In the field of pattern recognition, the task of grouping a set of objects based on their similarity is defined as *data clustering* [1]. In our problem, urban water consumption time-series data collected during one civil year, at a 1-minute time interval, was used.

Determining the most appropriate groups (clusters) and set of techniques for a new problem instance is not an easy task, as the same techniques can produce different results when applied on different data.

In our approach, different techniques were applied at distinct stages of the clustering problem, comprising the methodology presented in the next section.

3 Methodology

On a global perspective four main steps compose the proposed approach: (a) data pre-processing; (b) data representation; (c) data segmentation; and (d) test and validation. Figure 1 presents this methodology in an algorithmic structure.

Algorithm 1 Work Methodology

1) Data Pre-Processing
   a) Compute Unit of Analysis
   b) Impute Missing and Invalid Values
2) Data Representation
   a) Raw time-series
   b) Principal Components Analysis
3) Data Segmentation
   a) Hierarchical + K-Means Clustering
   b) TADPole Algorithm
4) Cluster Assessment
   a) Sum of Squared Errors
   b) Average Within-Cluster Sum of Squared Errors
   c) Silhouette Coefficient
   d) Calinski Harabaz Index

Figure 1: Work Methodology

3.1 Data Pre-Processing

Within the scope of the current work, data pre-processing comprises two main subtasks: determining the time-series' unit of analysis and imputing its invalid and missing values. These two subtasks must

be performed in this order (imputation of time-series can only be conducted after the unit of analysis has been computed).

3.1.1. Unit of Analysis

When identifying patterns of recurrent behaviour in time-series it is important to determine the periods of time with higher contributions to the overall signal. In other words, the most appropriate time horizon for clustering must be determined.

Similar works [2] [3] conduct a Fourier analysis by means of a *Fast Fourier Transform (FFT)*. Thus, the FFT was applied to the whole data set, highlighting a frequency approximating the 24h period as the (non-zero) frequency with the largest contribution to the signal, suggesting this to be our unit of analysis.

Based on this finding, an aggregation of the time-series values was conducted: objects to be grouped consisted in 1-day consumption records, with an interval of 1h.

3.1.2. Missing and Invalid Data Imputation

In almost all real-life applications and scenarios, errors in the data collection procedures are likely to occur, forcing data analysis and processing methods to address them properly. Substantial research has been performed on the topic of missing data imputation, resulting in different methods and techniques being proposed and tested in a variety of problems.

Steffen Moritz *et al.* [4] studied the application of different missing data imputation methods on univariate time-series. In their work, the authors obtained interesting results for approaches exploring linear interpolation, ARIMA and SARIMA models and Kalman Filters.

Alternative techniques can also be considered, ranging from traditional time-series analysis methods such as Linear interpolation and autocorrelation and trend analysis [2], to methods such as Artificial Neural Networks, Decision Trees, or K-Nearest Neighbours [5] [6].

Inspired by the results reported on the literature, an initial study was conducted comparing the performance of the following imputation methods: (i) linear interpolation; (ii) ARIMA; and (iii) Kalman Filter. Taking into account our unit of analysis, artificial invalid and missing values were injected in our data set. Each method was then applied to the individual samples and the correctness of their imputations was assessed using the *root mean squared error* metric.

Overall, the Kalman Filter provided better estimates than the competing alternatives, leading to estimates closer to the real values and also to better estimates in a higher number of days. Considering such results, this technique was selected to provide estimates for invalid and missing values in the considered data set.

3.2 Data Representation

When working with high dimensional data, computational requirements grow both in terms of memory and execution time.

Aghabozorgi *et al.* [7] points out another motivation for seeking less demanding representations: when measuring the distance between raw (high-dimensional) time-series samples, highly unintuitive results may be gathered resulting in the clustering of series similar in noise, instead of shape. The adopted distance metric also has a high influence on computed clusters.

Despite their computational burden, raw time-series representations remain quite popular in the literature. *Dynamic Time Warping* (DTW) [8] has strongly contributed for this popularity. Considered a more robust distance metric, the DTW has been extensively adopted with this representation.

Time-series representations based on dimensionality reduction techniques have also been proposed, of which *Principal Components Analysis (PCA)* [2] is an example. In PCA an orthogonal transformation

is performed, converting time-dependent observations in a set of values of linearly independent variables. By doing such, the temporal sequence of the data is lost and metrics such as the DTW cannot be employed. Alternative metrics, such as the Euclidean distance, need to be considered when applying these techniques to time-series.

According to [9], data normalisation techniques must also be carefully chosen. The authors claim that when working with raw time-series, performing a Z-Normalisation of the data can significantly improve the DTW's distance computation results.

Overall, two distinct time-series representation approaches were compared: (i) Raw time-series representation, featuring the popular DTW distance metric and a Z-Normalisation; and (ii) "Dimensionality-Reduced" time-series representation, obtained by means of PCA, adopting the Euclidean distance metric and a Min-Max Normalisation.

3.3 Data Segmentation

The choice of segmentation technique to apply in a clustering problem can strongly impact computed clusters. It is common to characterise and classify clustering approaches based on how data is grouped. Aghabozorgi *et al.* [7] distinguishes Hierarchical, Partitioning, Model-based, Density-based, Grid-based, and Multi-step clustering.

From the literature survey on time-series clustering Hierarchical and Partitioning techniques were identified as the most popular choice, with Hierarchical and K-Means Clustering the highlighted algorithms of these techniques, often applied together [10].

One of the main drawbacks of most time-series clustering techniques is related to the computational burden of finding similarities in the data. As the most robust similarity measures rely on a raw representation, high computational costs can be expected.

Motivated by the fact that equally robust and less demanding alternatives to the DTW metric have not yet been achieved, researchers have sought to reduce DTW's computational burden. The TADPole [11] algorithm is a popular example, which exploits upper and lower bounds on the DTW to drastically reduce the number of computations of this metric.

Given the dependence of TADPole on the DTW metric, this algorithm requires a raw time-series representation. Conversely, algorithms such as Hierarchical and K-Means clustering can be carried out on both considered representations.

Overall, the combination of Hierarchical and K-Means algorithms and the TADPole algorithm were applied to our data set, and their segmentation results were compared.

In the remainder of this subsection, details regarding the implementation and experimental setup of these two algorithms are presented.

3.3.1. Centroid Computation in K-Means

Besides specifying the target number of clusters, K-Means allows different centroid computation methods. A widely used method involves computing, for each dimension, the *average* of all samples assigned to the cluster in question.

When working with time-series, the average centroid computation method tends to be applied only for equal length series and a none-elastic distance metric (such as the Euclidean distance) [1].

When considering time-series of different lengths, or when distance metrics of other natures are employed (namely the DTW) simply performing the mean of the time-series at each point can fail to capture the average shape of all the time-series in a given cluster. As a result, alternative centroid computation methods more adequate to these distance metrics have been studied.

The *medoid* is a popular alternative to the average method, for raw representations. The medoid of the cluster is the sample that minimizes the sum of squared distances to all the other samples within the

cluster. However, this method does not appear to be a recurrent choice among researchers when dimensionality reduction techniques are adopted.

A more recent centroid computation method is the *DTW Barycenter Averaging* (DBA) [12]. This method was specified for the DTW metric and is claimed to outperform other centroid computation techniques when applied to UCR Archive's datasets. DBA seeks to minimise the sum of squared DTW distances from the average sequence (that is, the centroid) to all the sequences assigned to that group. A local optimisation strategy is implemented, with a strong dependency on the initial centroid guess.

Despite their popularity and documented ability to produce adequate cluster structures, both the DBA and medoid centroid computation methods lead to inadequate cluster structures for our consumption data set. A graphical evaluation revealed considerably inadequate centroids, poorly related to the consumption profiles assigned to each cluster. Such findings were also supported by the application of evaluation metrics described in Section 3.4.

As a result, concerning the centroid computation methods applied in the Hierarchical + K-Means combination only the *average* method was employed, for both raw and PCA representations.

3.3.2. TADPole

The TADPole algorithm requires the specification of a window size and a cut-off distance. TADPole assesses samples' similarity by computing the DTW distance in centred windows of a fixed size.

Upper and lower bounds on the DTW are explored to reduce its computation time and find time-series with many close neighbours. A cut-off distance is used as a threshold to determine time-series neighbours. Time-series that lie in dense areas are taken as the cluster centroids.

Initial experiments were performed with different values for this threshold, obtaining a value of *1.5*. A window size of 23 was defined, although further research on this value can be performed.

3.4 Test and Validation

Evaluating algorithms' performance in unsupervised learning problems is a challenging task and is still considered an open research problem [1], mostly because of the ambiguity and subjectivity of the cluster definition.

In unsupervised problems evaluation metrics involving *internal* indexes are applied, with cluster quality summarised to a single score without resorting to any labels or ground truth. Several internal indexes have been proposed in the literature, with the following being adopted in the current work:

- Sum of Squared Errors (SSE) [7]. In this context, the *error* of a sample is its distance to its cluster centroids. As clusters are desirably as dense as possible, smaller values of this metric suggest a more adequate cluster structure.
- Average Within-Cluster Sum of Squared Errors (AWCSS) [7]. This metric computes the average dissimilarity of samples belonging to the same cluster. Such a metric is necessary for the *elbow method*, a graphical inspection technique used to select the number of clusters for a given dataset.
- Silhouette Coefficient (SC) [13]. SC measures how similar a sample is to others in its own cluster in comparison to other clusters. SC takes a value in the range [-1, 1] where values closer to *1* suggests more dense and well-separated clusters:

$$s(i) = \frac{b(i) - a(i)}{\max\{b(i); a(i)\}} \quad (1)$$

- Calinski Harabaz Index (CH) [14], also referred to as the *Variance Ratio Criterion* (VRC). CH is unbounded and best suited for Euclidean distances: higher values of CH signal more dense and well-separated clusters:

$$CH = \frac{SS_B}{SS_W} \times \frac{(N-K)}{(k-1)} \quad (2)$$

Where SS_B and SS_W are the between and within-cluster variances, respectively, k is the number of clusters and N is the number of samples.

3.5 Experimental Setup

Figure 2 highlights the representation and segmentation techniques employed in this work.

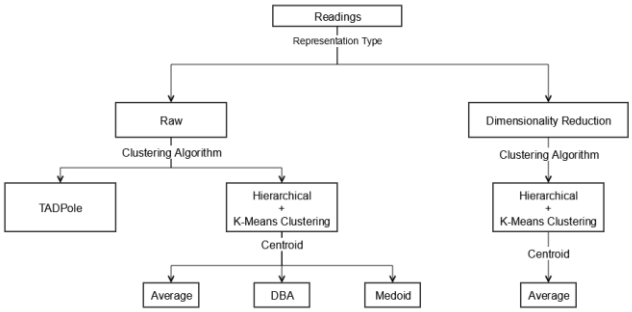

Figure 2: Experimental Setup

4 Results and Discussion

With respect to data representation and segmentation techniques, the following strategies were compared: (i) raw representation and the TADPole algorithm; (ii) raw representation and the Hierarchical and K-Means clustering algorithms; and (iii) PCA representation and the Hierarchical and K-Means clustering algorithms.

Despite its lightweight approach, the TADPole algorithm was unable to compute adequate cluster structures. Registered SC values were mostly negative or, when positive, very low. Hierarchical and K-Means clustering algorithms enabled more adequate cluster structures. As a result, Table 1 **Evaluation of the clusters computed with the Hierarchical and K-Means clustering algorithms.** presents the evaluation of consumption groups computed with the Hierarchical and K-Means algorithms.

Table 1 Evaluation of the clusters computed with the Hierarchical and K-Means clustering algorithms.

Data Representation	Number Clusters	SC	CH	SSE	AWCSS
Time-Series	2	0.855	702.506	2738.524	7.482
Time-Series	3	0.493	626.264	1279.169	3.495
Time-Series	4	0.438	452.708	1218.914	3.330
Time-Series	5	0.330	500.096	792.516	2.165
Time-Series	6	0.325	390.614	773.739	2.114
Time-Series	7	0.313	385.900	630.291	1.722
Time-Series	8	0.314	321.103	639.261	1.747
PCA	2	0.820	793.033	89.561	0.245
PCA	3	0.522	736.965	56.257	0.154
PCA	4	0.601	1351.579	23.333	0.064
PCA	5	0.601	1210.748	19.748	0.054

PCA	6	0.582	1106.588	17.391	0.048
PCA	7	0.446	1195.557	13.568	0.037
PCA	8	0.562	983.494	14.072	0.038

Analysing the elbow plot for the PCA representation, its elbow is located at four clusters. For the time-series representation, the elbow appears to be between three and five clusters. A closer analysis of the computed consumption groups with both representations revealed that three main consumption groups were always being computed:

- *Residual Consumptions*, predominant during summer months (June and July), characterised by low consumptions throughout the entire day (Figure 3, cluster 0).
- *Intense Consumptions*, predominant during warm months (July - September), characterised by high consumptions during business hours: 7am – 8pm (Figure 3, cluster 1).
- *Intense Consumptions*, predominant during colder months with some expression around the year. (Figure 3, cluster 2).

Despite the similarities between both intense consumption groups, their consumption profiles diverge during the night period. Between 3am and 6am lower water consumptions are registered during warmer months (Figure 3, cluster 1). The fact that consumption groups with similar profiles were obtained for both representations is a relevant result. Even though raw representations are more popular, PCA groups sets of few linearly independent variables, resulting in a smaller computation cost.

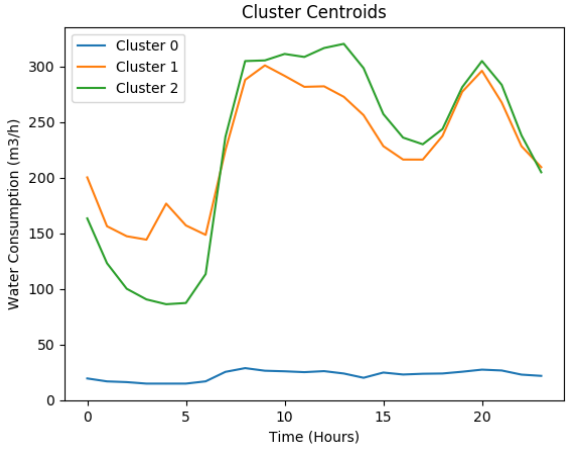

Figure 3 Consumption groups (Time-Series representation and Hierarchical and K-Means combination)

5 Conclusion

The current work categorises 1-year of urban domestic water consumptions. Results show that it is possible to obtain an automatic identification of recurrent behaviours and pinpoint time-dependent patterns on annual time-series. The ability to perform such categorisation is important to water companies, enabling them to manage their water supply infrastructures more accurately and efficiently.

Different techniques were applied to all stages of our clustering problem: data pre-processing, representation, segmentation and cluster assessment. Overall, three consumption profiles were computed: one representing residual and the remaining two representing intense consumptions, similar throughout the day except at night periods.

On a final remark, further extensions to this work can be considered by adopting nonlinear and more robust data representation techniques, as well as density-based segmentation strategies.

Acknowledgements

Joaquim Leitão gratefully acknowledges the Portuguese funding institution FCT – Foundation for Science and Technology –, Human Capital Operational Program (POCH) and the European Union (EU) for supporting this research work under the Ph.D. grant SFRH/BD/122103/2016.

References

[1] A. K. Jain, R. P. W. Duin and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 22.1, pp. 4-37, 2000.

[2] J. M. Abreu, F. C. Pereira and P. Ferrão, "Using pattern recognition to identify habitual behavior in residential electricity consumption," *Energy and Buildings,* vol. 49, pp. 479-487, 2012.

[3] F. Calabrese, J. Reades and C. Ratti, "Eigenplaces: segmenting space through digital signatures," *IEEE Pervasive Computing,* vol. 9.1, pp. 78-84, 2010.

[4] S. Moritz and e. al., "Comparison of different Methods for Univariate Time Series Imputation in R," *arXiv preprint arXiv:1510.03924,* 2015.

[5] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein and R. B. Altman, *Bioinformatics,* vol. 17.6, pp. 520-525, 2001.

[6] F. Yu and X. Xu, "A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network," *Applied Energy,* vol. 134, pp. 102-113, 2014.

[7] S. Aghabozorgi, A. S. Shirkhorshidi and T. Y. Wah, "Time-series clustering–A decade review," *Information Systems,* vol. 53, pp. 16-38, 2015.

[8] S. Chu, E. Keogh, D. Hart and M. Pazzani, "Iterative Deepening Dynamic Time Warping for Time series," in *Proceedings of the 2002 SIAM International Conference on Data Mining*, Washington, DC, USA, 2002.

[9] A. Mueen and E. Keogh, "Extracting optimal performance from dynamic time warping," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Minin*, San Francisco, CA, USA, 2016.

[10] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters,* vol. 31.8, pp. 651-666, 2010.

[11] N. Begum, L. Ulanova, J. Wang and K. Eamonn, "Accelerating Dynamic Time Warping Clustering with a Novel Admissible Pruning Strategy," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia, 2015.

[12] F. Petitjean, A. Ketterlin and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern Recognition,* vol. 44.3, pp. 678-693, 2011.

[13] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics,* vol. 20, pp. 53-65, 1987.

[14] T. Caliński and J. Harabasz, "A dendrite method for cluster analysi," *Communications in Statistics-theory and Methods,* vol. 3.1, pp. 1-27, 1974.