Space Time Series Clustering: Algorithms, Taxonomy, and Case Study on Urban Smart Cities

Asma Belhadi, Youcef Djenouri, Kjetil Nørvåg, Heri Ramampiaro, Florent Masseglia, Jerry Chun-Wei Lin

Space Time Series Clustering: Algorithms, Taxonomy, and Case Study on Urban Smart Cities

Asma Belhadi[1], Youcef Djenouri[2], Kjetil Nørvåg[3], Heri Ramampiaro[3], Florent Masseglia[4], Jerry Chun-Wei Lin[5]

[1] *Dept. of Technology, Kristiania University College, Oslo, Norway*
[2] *Dept. of Mathematics and Cybernetics, SINTEF Digital, Oslo, Norway*
[3] *Dept. of Computer Science, NTNU, Trondheim, Norway*
[4] *INRIA, LIRMM, Montpellier, France*
[5] *Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway*
asma.belhadi.17@gmail.com, youcef.djenouri@sintef.no, noervaag@ntnu.no, heri@ntnu.no, florent.masseglia@inria.fr, jerrylin@ieee.org

Abstract

This paper provides a short overview of space time series clustering, which can be generally grouped into three main categories such as: hierarchical, partitioning-based, and overlapping clustering. The first hierarchical category is to identify hierarchies in space time series data. The second partitioning-based category focuses on determining disjoint partitions among the space time series data, whereas the third overlapping category explores fuzzy logic to determine the different correlations between the space time series clusters. We also further describe solutions for each category in this paper. Furthermore, we show the applications of these solutions in an urban traffic data captured on two urban smart cities (e.g., Odense in Denmark and Beijing in China).The perspectives on open questions and research challenges are also mentioned and discussed that allow to obtain a better understanding of the intuition, limitations, and benefits for the various space time series clustering methods. This work can thus provide the guidances to practitioners for selecting the most suitable methods for their used cases, domains, and applications.

Keywords: Data mining, Clustering, Space time series

1. Introduction

Recent advances in geolocation, partly as a result of GPS (Global Positioning System) support, has resulted in the creation of large volumes of data varied in time and space. Space time series is one of the most powerful representations for several domains applications including transportation J. Hu et al. 2018, health-care Xi et al. 2018, seismology Morales-Esteban et al. 2014 and climate science Y. Liu 2015. The useful way to analyze space time series is by utilizing data mining and machine learning techniques Izakian, Pedrycz, and Jamal 2013; Von Landesberger et al. 2016. Clustering is one of the data mining techniques where similar data are grouped together into homogeneous clusters that has been intensively studied in the past decades Karypis, E.-H. Han, and Kumar 1999; Ng and J. Han 2002; Vesanto and Alhoniemi 2000; Karypis 2002; Nanopoulos, Theodoridis, and Manolopoulos 2001; H. Xiong, J. Wu, and J. Chen 2009; Ester et al. 1996; Jain, Murty, and Flynn 1999; Kriegel, Kröger, and Zimek 2009. In recent decades, many research focused on time series clustering Keogh and J. Lin 2005; Hallac et al. 2017; Dau, Begum, and Keogh 2016; Y. Xiong and Yeung 2002, and several works considered the spatial dimension in time series clustering Ferstl et al. 2017; Gharavi and B. Hu 2017; Izakian, Pedrycz, and Jamal 2015; Izakian, Pedrycz, and Jamal 2013, resulting in space time series clustering.

This paper presents a comprehensive overview of the existing space time series clustering algorithms. We have divided the existing approaches into three main categories depending on the type of clustering results. The first category is called hierarchical space time series clustering that is used to create hierarchical clusters among the space time series data. The second category is named pure partitioning space time series clustering that is utilized to partition the space time series into disjoint and similar clusters. For the third overlapping partitioning space time series clustering, it aims at determining clusters where space time series data may belong to one or more clusters. In this paper, we then study and present the solutions for each category. In addition, we show the applications of existing space time series clustering on urban traffic data relevant to two smart cities (e.g., Odense in Denmark and Beijing in China). Furthermore, challenges, open perspectives and research trends for space time series clustering are discussed and concluded. Compared to previous survey papers, this paper first provides

a deep analysis of space time series clustering techniques, which allows to clearly understand the merits and the limits of the reviewed algorithms for each space time series clustering category. This paper also derives mature solutions for space time series clustering, in particular for massive data, and for emerging applications.

1.1. Previous studies

This section summarizes the relevant survey papers and clarifies the differences to show the contributions of this paper. This survey paper is composed of two main topics, which are spatio-temporal data mining and time series clustering. In the following section, we review some existing surveys of these topics. Many data mining approaches have been proposed for spatio-temporal data.

Zheng 2015 reviewed trajectory data mining techniques including clustering, classification, and outlier detection. Feng and Zhu 2016 proposed an overall framework of trajectory data mining including preprocessing, data management, query processing, trajectory data mining tasks, and privacy protection. Shekhar, Evans, et al. 2011 and Gupta et al. 2014 provided the comprehensive overviews of application-based scenarios for spatio-temporal data mining such as financial markets, system diagnosis, biological data, and user-action sequences. Eftelioglu et al. 2016 studied hot spot detection in several applications such as environmental criminology, epidemiology, and biology. Keogh and Kasetty 2003 introduced the need of a fair evaluation of time series data including time series clustering. According to the authors, such as evaluation is done to avoid data and implementation bias. Liao 2005 presented an overview of time series clustering. It categorizes time series clustering into three categories, which are i) raw-data-based approaches either in time or frequency domain; ii) feature-based approaches that use feature extraction techniques for handling high dimensional time series reduction; and iii) model-based approaches that each time series is obtained by applying some mixture of models. Zolhavarieh, Aghabozorgi, and Teh 2014 reviewed the existing works of subsequence time series clustering based on the published periods such as: preproof (1997–2003), interproof (2003–2010), and postproof (2011–2014).

Fu 2011 discussed time series data mining techniques including segmentation, indexing, clustering, visualization and pattern discovery. Esling and Agon 2012 reviewed the existing time series data mining approaches such as classification, clustering, segmentation, outlier detection, prediction, and

rules and motifs discovery. It classifies time series clustering into two categories, which are i) whole series clustering by considering the complete time series in the clustering process, and ii) sub-sequences clustering, in which the clusters are found by selecting subsequences from multiple time series. In addition, Aghabozorgi, Shirkhorshidi, and Wah 2015 included another category of time series clustering, namely *time point clustering*, which aims at determining clusters based on a combination of the temporal proximity of time points and the similarity of the corresponding values. Compared to the existing surveys, this is the first survey that deals with space time series data; all the other works have been limited to only time series data, or even to spatial or temporal data.

Class	Algorithms	Variants
Hierarchical	Agglomerative Clustering	Rodriguez and Laio 2014 Shen and Cheng 2016 X. Wang et al. 2019
	Hierarchical Self Organizing Map	Steiger, Resch, and Zipf 2016 Y. Wu et al. 2017
	Machine Learning	Ferstl et al. 2017 Deng et al. 2018
Pure Partitioning	kmeans	N. Andrienko and G. Andrienko 2013 Gharavi and B. Hu 2017 Cho et al. 2014 H.-L. Yu et al. 2015 X. Jiang, C. Li, and J. Sun 2018 Bai et al. 2014 Krüger et al. 2017
	PAM	Von Landesberger et al. 2016 Penfold et al. 2016 T. Sun et al. 2017
	Peak Density	J. Jiang, Y. Chen, et al. 2019 Putri et al. 2019 Heredia and Mor 2019 H. Li 2019
Overlapping Partitioning	Fuzzy kmeans	Izakian, Pedrycz, and Jamal 2013 Izakian, Pedrycz, and Jamal 2015 Disegna, D'Urso, and Durante 2017
	Machine learning	Paci and Finazzi 2017 Gholami and Pavlovic 2017 Y. Zhang et al. 2017

Table 1: Taxonomy of space time series clustering algorithms

1.2. Taxonomy and paper organization

Table 1 presents a taxonomy of the space time series clustering algorithms presented in this paper. They are classified into three categories. The first

category is named *hierarchical clustering*, which is utilized to identify hierarchy among the space time series data. The second *pure partitioning space time series clustering* category is to partition the space time series into disjoint and similar clusters. Furthermore, the third *overlapping partitioning space time series clustering* category aims at determining a space time series that may belong to one or more clusters.

The rest of the paper is organized as follows. Section 2 defines the background and concepts used in the paper, including clustering and space time series data. Section 3 presents the relevant approaches for space time series algorithms. Section 4 shows a case study of the existing space time series clustering algorithms on large and big urban traffic data by exploring two urban smart cities (Odense in Denmark and Bejing in China). Section 5 discusses the challenges and future directions in space time series clustering. Finally, Section 6 states the conclusion of this paper.

2. Preliminaries

This section presents preliminaries regarding clustering techniques and space time series data.

2.1. Clustering

Definition 1 (Clustering). Consider m data $x_1, x_2, \ldots, x_m$, and a set of k clusters $C = \{C_1, C_2, \ldots, C_k\}$, and a distance measure D. Each cluster C_i is represented by its centroid g_i. Any clustering algorithm aims to partition the data into similar groups such as the optimal clustering denoted as C^*:

$$C^* = E_C \sum_{i=1}^{k} \sum_{x_j \in C} D(g_i, x_j) \tag{1}$$

Definition 2 (Hierarchical Clustering). Hierarchical clustering aims to create a tree-like nested structure partition $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2 \ldots \mathcal{H}_h\}$ of the data such that:
$\forall(i,j) \in [1 \ldots k]^2, \forall(m,l) \in [1 \ldots h]^2, C_i \in \mathcal{H}_m, C_j \in \mathcal{H}_l, m \geq l \Rightarrow C_i \in C_j \wedge C_i \cap C_j = \emptyset$

A hierarchical algorithm builds the hierarchical relationship among data. The typical approach is that each data point is first in an individual cluster. Based on the most neighboring, the clusters are merged to new clusters

until there is only one cluster left. Algorithms of this kind of clustering include BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) T. Zhang, Ramakrishnan, and Livny 1996, CURE (Clustering Using REpresentatives) Guha, Rastogi, and Shim 1998, ROCK (RObust Clustering Hierarchical) Guha, Rastogi, and Shim 2000, and Chameleon Karypis, E.-H. Han, and Kumar 1999.

Definition 3 (Pure Partitioning Clustering). Pure partitioning clustering aims to look for k partitions of the data such that:
$C_i \cap C_j = \emptyset$, *and* $\forall i \in [1 \dots k], C_i \neq \emptyset$

The basic idea of pure partitioning clustering is to consider the center of data points as the center of the corresponding cluster, and recursively compute and update the center until convergence criterion is achieved. Typical algorithms of this kind of clustering include k-means MacQueen et al. 1967, PAM (Partition Around Medoids) Kaufman and Rousseeuw 1990, and CLARA (Custering LARge Applications) Kaufman and Rousseeuw 2009.

Definition 4 (Overlapping Partitioning Clustering). Overlapping partitioning clustering indicates that each data x_j to each cluster C_i is with a degree of membership $\mu_{ij} \in [0 \dots 1]$ *such that* $\sum_{i=1}^{k} \mu ij = 1$

The basic idea of overlapping clustering is to assign data point to each cluster using a membership value between 0 and 1 in order to describe the relationship between data points and clusters. Typical algorithms of this kind of clustering include fuzzy c-means Bezdek, Ehrlich, and Full 1984, and FCS (Fragment Clustering Schemes) Dave and Bhaswan 1992.

2.2. Space Time Series

Definition 5 (Space Time Series). Consider m data such that $x_1, x_2, \dots, x_m$, each of data is comprised of a spatial part and a time series part. For the l^{th} data x_l, the concatenation $x_l = [x_l(s)|x_l(t)]$ is realized, where $x_l(s)$ represents its spatial part and $x_l(t)$ refers to its time series part. By considering r features (usually, $r = 2$) for the spatial part and q features for the time series part, we have the following representation for the l^{th} data with dimensionality $n = r + q$

$$x_l = [x_{l1}(s), \dots, x_{lr}(s)|x_{l1}(t), \dots, x_{lq}(t)] \tag{2}$$

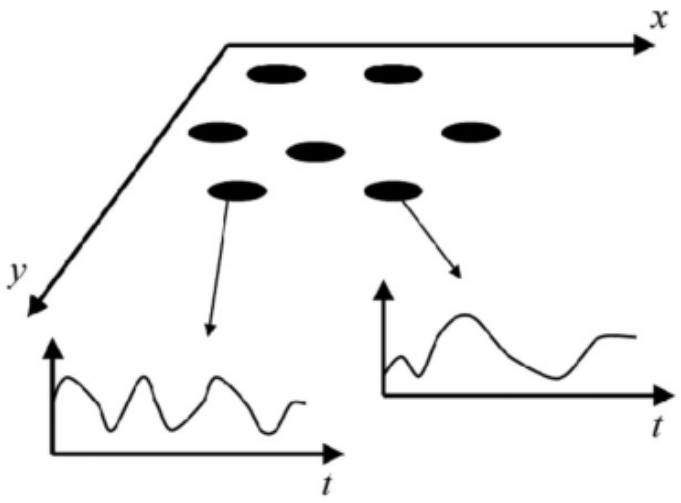


Figure 1: Space-time series example

Figure 1 illustrates an example of a space-time series. There are a number of spatial points in x-y coordinates, and for each spatial point, there is one (or more) time series representing the measurements of a phenomenon in different time steps.

Definition 6 (Space Time Series Similarity). We define the distance between two space time series x_1, and x_2 as:

$$D(x_1, x_2) = SD(x_1, x_2) + TD(x_1, x_2), \tag{3}$$

where i) $SD(x_1, x_2)$ defines the spatial distance that computes the similarity between the spatial components of the space time series x_1, and x_2, and ii) $TD(x_1, x_2)$ defines the temporal distance that determines the similarity between the time series components of x_1, and x_2. The spatial distance is usually computed using ordinary Euclidean distance, where the temporal distance is captured using time series distances Mori, Mendiburu, and Lozano 2016. In the following, we illustrate some interesting measures between two time series data (e.g., $x(t)$ and $y(t)$).

1. L_p distances Yi and Faloutsos 2000: L_p distances are the rigid metrics that can only compare series of the same length. However, due to their

simplicity, they have been widely used in many tasks related to time series analysis and mining. The different variations of the L_p distances and their formulas are provided in Table 2.

Table 2: L_p Distances

Distance	p	Formula
Manhattan	1	$\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)} \lvert x_i(t) - y_i(t)\rvert$
Euclidean	2	$\sqrt{\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)}(x_i(t) - y_i(t))^2}$
Minkowski	$[1 \dots \infty]$	$\sqrt[p]{\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)}(x_i(t) - y_i(t))^{1/p}}$
Infinite norm	∞	$\max_{x_i(t)\in x(t)\vee y_i(t)y(t)}\{\lvert x_i(t) - y_i(t)\rvert\}$

Table 3: Time Series Distances

Distance	Formula
DTW(x(t), y(t))	$\begin{cases} 0 & \text{if } \lvert x(t)\rvert - 1 = \lvert y(t)\rvert - 1 = 0 \\ \infty & \text{if } \lvert x(t)\rvert - 1 = 0 \lvert\lvert \lvert y(t)\rvert - 1 = 0 \\ x_0(t) - y_0(t) + min\{DTW(x(t)/x_0(t), & \\ y(t)/y_0(t)), DTW(x(t), y(t)/y_0(t)), & \text{otherwise} \\ DTW(x(t)/x_0(t), y(t))\} & \end{cases}$
STS(x(t), y(t))	$\sqrt{\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)} \left(\frac{y_{i+1}(t)-y_i(t)}{t_{i+1}-t_i} - \frac{x_{i+1}(t)-x_i(t)}{t_{i+1}-t_i}\right)}$
Dissim(x(t), y(t))	$\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)}(x_i(t) - y_i(t) + x_{i+1}(t) - y_{i+1}(t)) \times (t_{i+1} - t_i)$
PC(x(t), y(t))	$\frac{\sum_{x_i(t)\in x(t)\vee y_i(t)y(t)}(x_i(t)-\overline{x(t)})\times(y_i(t)-\overline{y(t)})}{\sqrt{(x_i(t)-\overline{x(t)})^2}\times\sqrt{(y_i(t)-\overline{y(t)})^2}}$ Note that $\overline{x(t)}$ and $\overline{y(t)}$ are the mean values of time series $x(t)$ and $y(t)$, respectively.

2. DTW (Dynamic Time Warping) Sakoe and Chiba 1978: This distance is able to deal with transformations such as local warping and shifting. Furthermore, it allows the comparison between different series length.
3. STS (Short Time Series) Möller-Levet et al. 2003: This distance is adapted to the characteristics of irregularly sampled series.
4. Dissim Frentzos, Gratsias, and Theodoridis 2007: It is specifically designed for series collected at different sampling rates that indicates each series is defined in a finite set of time instants, but these can be different for each series.
5. PC (Pearson‘s correlation) Golay et al. 1998: This distance focuses on extracting a set of features from the time series and calculating the similarity between these features instead of using the raw values of the series.

Table 3 presents the description of DTW, STS, Dissim, and PC formulas. The interested readers may also find the alternative distances with

SAX (Symbolic Aggregate approXimation) Cole, Shasha, and Zhao 2005 or sketches Shieh and Keogh 2008 but until now, we do not find those relevant works in space time series clustering.

3. Algorithms

Intensive studies have been carried to capture the space time series clustering algorithms. Up to now, 112 papers have been analyzed. From these papers, the following filter process is performed:

1. 28 papers are removed after the first pre-screening of the abstract due to they are out of the scope of space time series clustering.
2. From the remaining 84 papers, only 32 papers are finally selected. The selection criteria is based on the quality of the paper, and the quality of the publisher. Only high quality papers are elected if the paper is published in top-tier conferences or high impact factor journals.

In addition, our intensive study reveals that almost solutions for space time series clustering can be classified as hierarchical-based, pure partitioning-based, and overlapping-based approaches. Therefore, this section presents space time series clustering algorithms grouped into three categories.

3.1. Space-time series hierarchical clustering

Before 2017: Rodrigues, Gama, and Pedroso 2008 presented ODAC (Online Divisive Agglomerative Clustering) incrementally updates the tree-like hierarchy clusters using top-down strategy by computing the dissimilarities between time series. The dissimilarities between time series is defined by the cluster‘s diameter measure, where the splitting criterion is supported by a confidence level given by the Hoeffding bound Hoeffding 1963 that allows to define the diameter of each cluster. This algorithm is applied to multiple time series, but it can be easily adopted on space time series data. Shen and Cheng 2016 established a new framework that enables comprehensive analysis of trajectory space-time series data to group people with similar behavior. The framework segregates individuals into subgroups upon where (place), when (time) and how long (duration) the activities are conducted for each individual. It includes three main steps, (i) extracting ST-ROI, i.e., the region of interests with not only spatial location information but also interesting time spans; (ii) defining individual time allocation on the ST ROIs as their space-time profiles to describe his/her activity routine; and (iii) group

people using hierarchical clustering based on different activity patterns. For example, Steiger, Resch, and Zipf 2016 develop the geographic hierarchical self-organizing map (Geo-H-SOM) for discovering spatio-temporal semantic clusters from tweets of people generated in different week days and on different regions. It considers the similarities between tweets across their temporal as well as geographical and semantic characteristics. Results on real case study in London reveal similar correlations between tweets of people live in the same region. However, some clusters are sparse and difficult to be analyzed due to high correlation between certain tweets.

From 2017: Ferstl et al. 2017 propose a hierarchical ensemble clustering approach to analyze and visualize temporal uncertainty in weather forecasting data. Clusters in specified time window are merged to indicate when and where forecast trajectories diverge. Different visualizations of time-hierarchical grouping on European center for medium-range weather forecasting data are shown including space-time surfaces built by connecting cluster representatives over time, and stacked contour variability plots. Y. Wu et al. 2017 presented an interactive framework named *StreamExplorer* that visualizes social streams. It continuously detects important time periods (i.e., sub-events), and extract topics of tweets made on any sub-events using GPU-assisted self organizing map. A multi-level visualization method that integrates Agnes algorithm for showing a space-time series generated from the extracted tweets in a given time period and for different users located on different regions. The map allows to summarize important sub-events at a macroscopic level using a tree of visualizations. It not only reveals the dynamic changes of a social stream in the context of its past evolution, but also organizes historical sub-events in a hierarchical manner for easy review and navigation of sub-events. The proposed system enables end users to track, explore, and gain insights of social streams at different levels. Deng et al. 2018 address a spatio-temporal heterogeneity problem by employing space-time series clustering. This approach divides space-time series data into meaningful clusters while considering both the spatial proximity and the time series similarity instead of the previous methods that only deal with time or space dimensions. The application of auto-correlation time-series clusters in artificial neural networks reveals good accuracy for space-time series prediction. X. Wang et al. 2019 suggested a novel representation formed by a sequence of 3-tuples for interval-valued time series in high dimensional data, and loss information issue. In addition, a hierarchical clustering algorithm based on improved dynamic time warping distance measure is designed for

interval-valued time series of equal or unequal length.

3.2. Space-time series pure partitioning clustering

Before 2016: N. Andrienko and G. Andrienko 2013 proposed an interactive framework to analyze and visualize large amount of spatio-temporal data represented by a set of time series. Multiple and heterogeneous space-time series are first created from the spatio temporal data. The set of space-time series is then grouped based on the similarity of the temporal variation of the timestamp value. During this step, an interactive tool is used to refine the clustering results. This is done by showing both time graph and map displays to the data analysts. The time-graph display let the analysts view the homogeneity degree of each group. For the map display, the locations are characterized by the space-time series and each group is painted in the same color. Cho et al. 2014 develop the Stroscope visualization tool that help neurologists analyze space-time series coming from blood pressure data. It provides two kinds of clustering techniques such as: data-space clustering and image-space clustering. For the data-space clustering, records that have similar measurement values are grouped together, which result in the same clusters for the same data set. However, an image-space clustering aims at solving the visual inconsistency problem of the data-space clustering. In the image space clustering, records with a similar color pattern are clustered together, where the clustering results could vary according to the color table defined by the users, but the results are more reasonable to users who expresses his/her intention in his/her color mapping choice. Bai et al. 2014 proposed a Gtem algorithm to cluster events from geographical temperature sequence data. It can detect high temperature events in irregular shape, size and evolution model. Furthermore, Gtem can automatically select the optimal parameters based on the MDL (Minimum Length Description) principle Barron, Rissanen, and B. Yu 1998 to automatically group events of an area into space-time series data. Moreover, Gtem can successfully find high-temperature events with exact start-end timestamps on the daily weather of the Hunan province in China from 2004–2008. H.-L. Yu et al. 2015 applied the clustering process to distinguish the space-time patterns of local precipitations in the summer and autumn synoptic conditions from 24 gauges during 1996–2008 in Taiwan. It groups the synoptic and local conditions for the space-time rainfall patterns by integrating k-means with the empirical orthogonal function analysis. The results identified three mainly extreme patterns and two normal patterns in both seasons. L. Li et al. 2015 in-

vestigated trend modeling for space time series clustering in the context of urban traffic data. Based on daily similarity of traffic time series on different urban locations, the simple average trend with PCA (Principle Component Analysis) approach is developed to analyze the daily traffic space time series obtained in consecutive days and define their global varying similarity while DWT (Discrete wavelet transform) mostly defines the local varying tendency.

2016: Von Landesberger et al. 2016 presented a visual analysis for people flow between places in London. The people flow is aggregated into regions to reduce the mass mobility patterns using k-means algorithm. Despite of visualization of people flow, only aggregated regions are shown to the users for better understanding the distribution of flow between places. Moreover, a new measure named *Strength Flow* is developed to filter the regions having low density flow. L. Wang et al. 2016 developed a clustering method called separation degree algorithm that is able to construct self-adaptive interval based on the separation degree model to detect anomaly in network space data. The advantage of this approach is to automatically determine the self-adaptive interval, which can be used to improve the accuracy of anomaly detection. Extensive experimental results showed that the proposed method can effectively detect anomaly data from heterogeneous spaces in the given network. Penfold et al. 2016 suggested a clustering model to identify clusters of early adoption for a new clinical practice. The results indicated that the revealed patterns provide insights to identify organizational context and prescribe level factors involved in diffusion and implementation within a learning health care system. The proposed approach can be used for real-time prospective surveillance context such as urgent clinical events with public health importance. Steiger, Resch, Albuquerque, et al. 2016 used a geographic self-organizing map to group human mobility patterns by analyzing similar space-time series generated from live traffic feeds. A standard self-organizing map is first applied in order to observe and analyze the general topological relationships of the reference database. A geographic self-organizing map is then computed for the identification of similar overlapping traffic disruption patterns. The results of traffic disruption clusters are finally correlated with the computed geo-referenced weight vectors from all retrieved geo-referenced traffic data. A case study in London traffic data showed that particularly special events, such as concerts, demonstrations, and sports events, etc., are well reflected within space-time series input data.

2017: T. Sun et al. 2017 proposed matching and pruning strategies to efficiently compute the center of space-time series using dynamic time warping

distance. Experimental results revealed that the proposed centroid formula improves the performance compared to the existing space-time series clustering in terms of computational time and clustering quality. Gharavi and B. Hu 2017 presented a clustering algorithm to detect disturbances and degradation area in the grid. The k-means algorithm is extended to group measurement units into different clusters based on power quality. A multi-objective criterion is defined by considering both time and space in the clustering process. According to the experiments on IEEE 39-bus transmission system, it revealed that the proposed clustering space-time synchrophasor scheme is capable to detect and isolate areas in the grid suffering from multiple disturbances, such as faults. Krüger et al. 2017 proposed a segmentation approach that allows distinct activities within human motion space-time series data. Segmentation human motion data is first considered as graph problem, and a neighborhood graph-based and PCA (Principle Components Analysis) approaches are then applied for dimension reduction. A clustering method that allows to detect motion segments is based on self-similarities which needs no assumption on the number of clusters. The experiments on a wide variety of motion datasets show that the approach can identify usual non-repetitive human activities such as one step, jump, and, turn. However, the approach could not identify some substantial changes between the individual repetitions in the muscle activation patterns such as fatigue effects in longer motion trials.

2018: X. Jiang, C. Li, and J. Sun 2018 focused on mining of multimedia time series data using a mixed composition of graphic arts and pictures, hyper text, text data, video or audio. It adopted a k-means algorithm to handle high dimensional data as the input set for a multimedia database and at the same time, the algorithms obtains optimal similarity measure by utilizing a Minkowski distance which is a generalized form of the Euclidean distance. Mikalsen et al. 2018 proposed a robust time series cluster kernel by taking the missing time series data into account using the properties of Gaussian mixture models augmented with informative prior distributions. An ensemble learning approach is exploited to ensure robustness of parameters by combining the clustering results of many Gaussian mixture models to form the final kernel.

2019: Some algorithms based on density peaks principle Rodriguez and Laio 2014 and its variants gravitation-based density peaks clustering J. Jiang, Hao, et al. 2018, and density peaks clustering based on logistic distribution and gravitation J. Jiang, Y. Chen, et al. 2019 have been suggested for space

time series clustering. The main idea behind these algorithms is that the centers of the different clusters are more dense than the remaining data. This allows to automatically identify outliers. In addition, the clusters are recognized regardless of their shape and of the dimensionality of the space in which they are embedded. Y. Wang et al. 2016 implements a time-based Markov model to formulate the dynamics of electricity consumption for customer behaviors by considering the state-dependent characteristics. It also indicates that future consumption behaviors would be related to the current states. Furthermore, it mentions that the density peak clustering has good robustness to identify outliers without further processing. Putri et al. 2019 developed a new density-based clustering approach for grouping a set of time series. This approach generates arbitrarily shaped clusters, and explicitly tracks their temporal evolution. Heredia and Mor 2019 proposed a hybrid approach which combines the density-peak clustering with the spatial density of space time series data. The whole data is first partitioned using the smoothed density function, and the resulted groups are further divided using the density-peak clustering approach. H. Li 2019 developed a hybrid approach based on principal component analysis and density-peak clustering. A high dimensional multi-variate space time series data is first reduced using the principal component analysis, and the selected features are then grouped using the density-peak clustering.

3.3. Space-time series overlapping partitioning clustering

Before 2017: Izakian, Pedrycz, and Jamal 2013 proposed a fuzzy approach for spatio-temporal data clustering. Fuzzy c-means and adaptive Euclidean distance function are adopted to cluster different nature of spatio-temporal data. The suggested augmented distance allows to control the effect of each data in the determination of the overall Euclidean distance and gives a sound balance between the impact of the spatial and temporal components of the data. Izakian and Pedrycz 2014b suggested a cluster-center approach for anomaly detection problem in space-time series data. A Fuzzy C-Means (FCM) algorithm is employed to group the time series. A Euclidean distance is used for similarity computation in both spatial and temporal components, where the λ parameter is defined to balance the impact of the spatial and temporal components of the data in the clustering process. In addition, Anomaly score is assigned to each cluster for quantifying the unexpected changes in the structure of data. At the end, the relations between clusters presented in successive time windows are visualized to quantify anomaly propagation over

time. Izakian and Pedrycz 2014a introduced a generalized version of fuzzy c-means clustering to cluster data with blocks of features coming from distinct sources. A new distance function is developed to take the multi-sources aspect into account. The distance combines the features of different sources by using aggregate variables, that allows to increase/decrease the impact of the given data source against other data sources. Izakian, Pedrycz, and Jamal 2015 further presented three alternative approaches for fuzzy clustering of space-time series data. The first approach takes the averaging dynamic time warping distance technique into account and applies the fuzzy c-means technique for clustering space-time series data. For the second approach, a fuzzy c-medoids technique that ignores the average distance calculation was explored and finally, a combination between the c-medoids and the c-medoids was examined and discussed.

From 2017: Paci and Finazzi 2017 developed a Bayesian dynamic approach that integrates a finite weighted mixture model for clustering space-time series data. Thus, a state-space model has been employed to describe the temporal evolution of different locations belonging to each cluster. Also, a new strategy for selecting the number of clusters has presented. By using a weighted mixture model, this approach allows easy and fast prediction of the membership probability at any location and at any window time. Disegna, D'Urso, and Durante 2017 proposed COFUST (COpula-based FUzzy clustering algorithm for Spatial Time series). A combination of Fuzzy Partitioning Around Medoids (FPAM) algorithm Kaufman and Rousseeuw 1990 with a copula-based approach Di Lascio, Durante, and Pappada 2017 is performed to interpret co-movements of large-scale time series. First, both spatial and temporal dependencies between the time series are identified through a copula-based approach. Then, the FPAM algorithm has been adopted in order to determine non-fictitious patterns in the space-time series and producing the final clusters. This approach is computationally more efficient and tend to be less affected by both local optima and convergence problems compared to the existing space-time series overlapping clustering. Gholami and Pavlovic 2017 considered the temporal dependency between space-time series of complex human motion data. The temporal dependencies are modeled using Gaussian process whose covariance function controls the desired dependence. The Bernoulli process is also incorporated into the overall process to concurrently learn the dimensionality of the subspaces from the data. Y. Zhang et al. 2017 introduced a density-contour based spatio-temporal clustering approach (ST-DPOLY) and compare it with the spatio-temporal

shared nearest neighbors (ST-SNN). First, a spatial density function is determined for the spatial point data collected in batches, where a density threshold is used for each batch of time to identify spatial clusters. Spatio-temporal clusters are then determined as continuing clusters. Continuing clusters are defined as the clusters highly correlated in consecutive batches of time. The proposed approach has applied to 1.1 billion taxi trips recorded over seven consecutive years from 2009 to 2016, and presented advantages in terms of clustering results, time and space complexity, while ST-SNN is more interesting in terms of temporal flexibility.

3.4. Discussions

From the above literature review, we provide our insights of the reviewed papers.

1. Clustering of space time series data requires a lot of efforts, especially in terms of a suitable treatment of the spatial and temporal components of the data. Existing space time series clustering algorithms have been developed in this direction. Still, much further work is needed to achieve mature solutions. For instance, current algorithms consider temporal and spatial dimensions in the same processing level. However, in some cases, temporal dimension is more suitable than the spatial one, and vice versa. One way to tackle this issue is to transform the space time series clustering as multi-objective optimization problem, where some aggregation functions may be used between the different dimensions (spatial and temporal dimensions in this case).
2. Space time series data are usually gathered from sensors, and should be processed continually in a data streams environment. The major concern with the existing clustering time series is that they do not provide mechanism to deal with data streams. Incremental clustering algorithms may be an alternative solution. The main merit of these algorithms is that it is not necessary to store the whole data in the memory. Thus, the space requirement of incremental algorithms is relevant small. Typically, they are non-iterative; their time requirement is also small. Adopting incremental clustering algorithms in a space time series is beneficial to practitioners for dealing with more real-world applications relevant to manufacturing and smart city, among others.

Table 4: Characteristics of space time series clustering algorithms.

Category	Merits	Limits
Hierarchical Clustering	Free-parameters Different level of granularities	High time and memory consuming
Pure Clustering	Fast time and memory consuming	Difficult to fix the number of clusters
Overlapping Clustering	Finding Overlapping clusters	High time consuming Need parameters adjustments

In addition, we present the merits and limitations of the existing space time series algorithms (See Table 4 for more details). We can classify the space time series algorithms into three groups, according to various clustering models:

1. Algorithms in the first group aim at finding hierarchical clusters. They do not need any parameter as the input (i.e., the number of clusters). In addition, it is possible to examine partitions at different granularity levels. However, with large scale data, they require higher computation and a huge memory usage. However, space time series data is normally large scale, thus this model is not suitable for real-world situations.
2. The purpose of the algorithms in second group is to find the disjoint partitions. These algorithms are fast compared to the algorithms in the first category. They are thus more suitable for large scale space time series data. Nevertheless, those algorithms require parameter setting (i.e., the number of clusters), which is normally hard to decide, in particular while considering more dimensions in space time series data.
3. Algorithms in the third group are overlapping partitioning algorithms, which are used to find the overlapping partitions by using a membership degree as input. These algorithms are slow compared to pure partitioning algorithms due to the complexity of the space time series data. Moreover, they are very sensitive to the membership rate and the number of clusters. In addition, to determine the overlapping clusters, they do not distinguish the spatial and temporal dimensions, which degrades the accuracy performance. In many real-world cases, some data may overlap in one dimension but are disjoint in the other. In such cases, overlapping partitioning algorithms would fail to determine the optimal clustering.

4. Evaluation

In this section, a performance evaluation of space time series clustering is provided. Both standard time series databases [1] and a real case study on urban traffic intelligent transportation are analyzed as follows.

4.1. Case Study: Urban traffic Intelligent Transportation

With the popularization of GPS and IT devices, urban traffic flow analysis has attracted growing attention in the last decades.Zheng 2015 and Feng and Zhu 2016 reviewed spatio-temporal urban data mining techniques. The surveys included segmentation and clustering, detecting outliers and anomaly flows, classification sub-trajectories, and finding frequent and periodical sequential patterns from clusters of trajectories. The traffic flow is computed by counting the number of objects (i.e., cars, passengers, taxis, buses, etc.) across a given location during a time interval. This generates a high number of time series captured in different locations of the urban city. A trivial way to represent these time series captured in different locations is space time series data. Space time series data mining is largely used in many number of domains related to intelligent transportation Jensen et al. 2016; Feng and Zhu 2016. They are used to adapt classical data mining techniques and propose new methods for discovering useful knowledge from urban traffic space time series data. Recent research works of space time series data mining techniques for urban traffic data including clustering, pattern mining, and outlier detection can be found in Shekhar, Evans, et al. 2011; Zhou, Shekhar, and Ali 2014; Koperski, Adhikary, and J. Han 1996; Gupta et al. 2014; Shekhar, Z. Jiang, et al. 2015; Y. Djenouri, Belhadi, J. C.-W. Lin, D. Djenouri, et al. 2019; Y. Djenouri, Belhadi, J. C.-W. Lin, and Cano 2019; Y. Djenouri and Zimek 2018; Y. Djenouri and Zimek 2018. One application of space time series data mining for urban data is clustering. The goal is to find out the similar clusters of urban traffic flows represented in different locations. This section shows a case study of an application of space time series clustering algorithms for dealing with urban traffic data. In the experiments, we consider various clustering algorithms with different similarity measures on two urban traffic data (large dataset for Odense city in Denmark and big dataset for Beijing city in China).

[1] https://archive.ics.uci.edu/ml/index.php

4.2. Datasets

Two real Odense and Beijing traffic flow data have been used for evaluation. These datasets are varied in terms of the number of flow values, The Odense traffic data is considered as a large dataset, where the Beijing traffic data is considered as big dataset. The detailed explanation of these two datasets is given as follows:

Odense Traffic Data: The first data is captured from several test locations throughout the Odense city. Each data entry contains information related to the vehicle or bike detected at specific locations such as: gap, length, date, time, speed, and class (i.e., type of vehicle or bike). The location is represented by latitude and longitude dimensions. The speed is calculated by km/h, and the datetime represents the year, the month, the day, the hour, the minute and the second that the vehicle or bike is passed by the given location. The most important information of each vehicle or bike is given as follows:

- datetime: It represents the time that the vehicle or bike passed on the location, and the format is: YYYY-MM-DD hh:mm:ss.
- speed: It defines the actual speed of the vehicle or bike where it is across the location.
- class: It defines the type of vehicle or bike. For example, if the class is set as 2, it represents a passenger car.

For ten locations, sensor infrastructure has been installed in a pilot experiment. The ten locations have different characteristics (i.e., traffic density, counters for cars/bikes) as described in Table 5. The traffic data were obtained between January 1^{st}, 2017 and 30^{th} April 2018. It consists of more than 12 million vehicles and bikes. The data is made by Odense Kommune [2], and may be retrieved at `http://dss.sdu.dk/projects/its/fpd-lof.html`.

Beijing Traffic Data: The second one is a real urban traffic data obtained from Beijing traffic flow, and retrieved from [3]. It consists of more than 3 billion traffic flow values during a two-months time period on more

[2]https://www.odense.dk/

[3]https://www.beijingcitylab.com/

Table 5: Odense Data Description

Address	ID	Type	#(Cars or Bikes)
Falen	L_1	Cars	16.932
Anderupvej	L_2	Cars	25.310
Aløkke Alle	L_3	Cars	238.775
Thomas B Thriges Gade	L_4	Bikes	46.978
Niels Bohrs Alle	L_5	Bikes	445.883
Rødegårdsvej Østgående	L_6	Bikes	575.089
Rugårdsvej	L_7	Cars	2.318.852
Nyborgvej	L_8	Cars	2.352.930
Grønlandsgade	L_9	Cars	2.955.464
Odins Bro	L_{10}	Cars	3.921.746

than one hundred locations. The most important information of each car is given as follows:

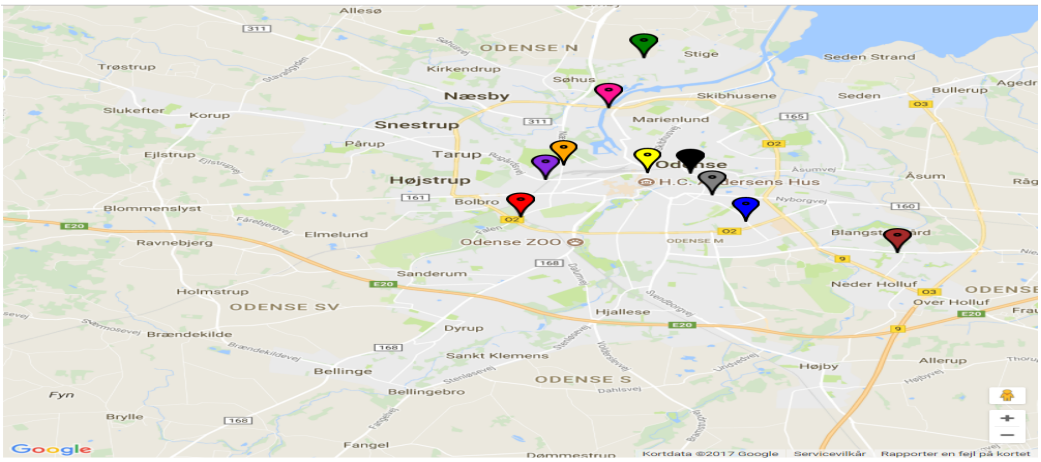
- datetime: It represents the time that the car passed on the location, and format is: YYYY-MM-DD hh:mm:ss.
- Class: It defines the type of vehicle or bus.

Figure 2 presents the distribution of urban traffic data among Odense and Beijing cities.

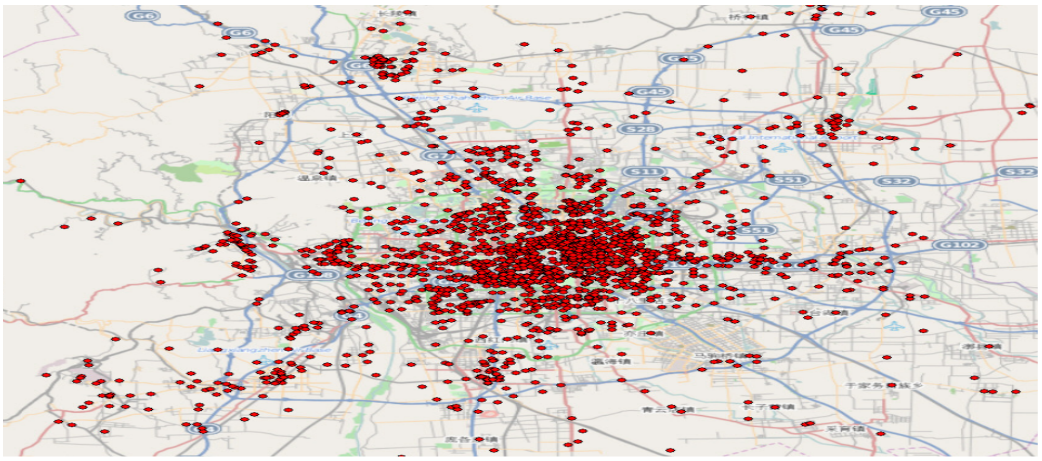
4.3. Tool

In this work, we used the algorithms that implemented in SPMF library [4] to perform the space time series clustering. The k-means, DBSCAN, fuzzy c-means, CHA, and Density-Peak are the well-known clustering algorithms, and cover all the categories of space time series clustering. Therefore, these algorithms are chosen for performance evaluation. This library is first proposed for pattern mining discovery Fournier-Viger et al. 2014 that has been extended to different data mining applications. It also provides algorithms for analyzing time series data such as SAX J. Lin, Keogh, Wei, et al. 2007 and PAA J. Lin, Keogh, Lonardi, et al. 2003, among others. SPMF is an open-source library implemented in Java. The current version of this tool is v2.42b and was released on 11-th, March, 2020. It currently contains 196 data mining algorithms. We then used SPMF for space time series clustering algorithms by implementing the distances related to space time series data, adding class to space time series representation, and adapting the existing

[4]http://www.philippe-fournier-viger.com/spmf/



Odense Locations



Beijing Locations

Figure 2: Urban Traffic Odense and Beijing Locations

clustering algorithms provided in SPMF library for space time series representation. We evaluate algorithms regarding three different dimensions as follows:

1. **Runtime performance**: We perform the runtime of each space time series clustering algorithm including preprocessing step, computing similarity, and determining clusters.
2. **Quality of clusters**: We evaluate the quality of the clusters by two ways. The first way aims to compute the Error Sum of Squares (ESS).

It is the sum of the squared differences between each space time series data and the mean of its group. It can be used as a measure for variation within a cluster. If all cases within a cluster are identical, the ESS is equal to 0. A better clustering result obtains lower ESS value. The ESS formula is given as:

$$ESS = \sum_{i=1}^{k} \sum_{x_j \in C_i} (x_j - \overline{C_i})^2, \tag{4}$$

where $\overline{C_i}$ is the mean value of the space time series data belonging to the cluster C_i.

The second way aims to evaluate the quality of clusters for the classification of traffic flow Sumit and Akhter 2019; Rezaei and X. Liu 2019; Qu et al. 2019. The data labels are created for each time series data based on the daily observed traffic. We have obtained three different labels (WD: data for weekday, ST: data for Saturday, and SN: data for Sunday). We have created two files; the first file contains the data without labels, and the second file contains data with labels. We apply space time series clustering techniques on the first file and set the number of clusters as 3 (for DBSCAN and CHA algorithms). We have adjusted their parameters to find 3 clusters in order to make a fair comparison with k-means, and fuzzy c-means algorithms. After construction of clusters, we compare each cluster for the data with the same label, and we compute the number of corrected classified data for each cluster as the maximum number of common data between this cluster and each label. We then computed the classification ratio of each algorithm to see evaluate the performance.

For both ways, we used the k-fold cross-validation technique, which is largely used in the machine learning community Anand, Kirar, and Burse 2013. This approach involves randomly dividing the set of observations into k groups or folds of approximately equal size. The first fold is treated as a validation set and the model is fit to the remaining folds. The procedure is then repeated k times, where a different group is treated as the validation set.

3. **Memory usage**: We compute the memory consumption of the space time series clustering algorithms by using the *MemoryLogger* provided in SPMF tool.

Table 6: Comparison of the space time series clustering in terms of runtime (seconds), the quality of returned clusters (ESS), and the memory usage (MB) using standard time series databases.

Algorithm	Dataset	CPU	ESS	Memory
kmeans	Air Quality	25	1.80	29
	Appliances energy prediction	29	2.21	31
	EEG Eye State	31	2.25	35
	Real-time Election	35	2.51	38
	Beijing Multi-Site Air-Quality	38	2.59	40
	Beijing PM2.5	42	2.71	45
DBSCAN	Air Quality	27	1.71	30
	Appliances energy prediction	31	2.25	33
	EEG Eye State	32	2.28	37
	Real-time Election	37	2.35	41
	Beijing Multi-Site Air-Quality	39	2.57	43
	Beijing PM2.5	41	2.95	47
Fuzzy cmeans	Air Quality	25	1.80	29
	Appliances energy prediction	33	2.57	39
	EEG Eye State	41	2.59	45
	Real-time Election	45	2.99	49
	Beijing Multi-Site Air-Quality	49	3.11	59
	Beijing PM2.5	55	3.05	52
CHA	Air Quality	42	0.99	40
	Appliances energy prediction	43	1.05	51
	EEG Eye State	49	1.12	56
	Real-time Election	56	1.19	62
	Beijing Multi-Site Air-Quality	63	1.68	66
	Beijing PM2.5	67	1.81	71
Density Peak	Air Quality	22	1.52	31
	Appliances energy prediction	31	1.71	30
	EEG Eye State	32	1.75	40
	Real-time Election	35	1.74	51
	Beijing Multi-Site Air-Quality	39	1.77	53
	Beijing PM2.5	45	1.92	54

4.4. Results on Standard Time Series Data

In this experiment, the evaluation of the space time series clustering is carried out on standard time series databases `https://archive.ics.uci.edu/ml/index.php`. Table 6 lists the runtime, the ESS value, and the memory usage for different used time series databases. From this table, we can observe that the k-means and DBSCAN are the most powerful methods compared to the other space time series clustering algorithms. Fuzzy c-means and Density Peak are less competitive than k-means and DBSCAN. However, they have obtained reasonable results compared to CHA. This latter is the less competitive algorithm, which requires high computational and memory resources, and it provides less quality of clusters.

4.5. Results on Urban Odense Traffic Data

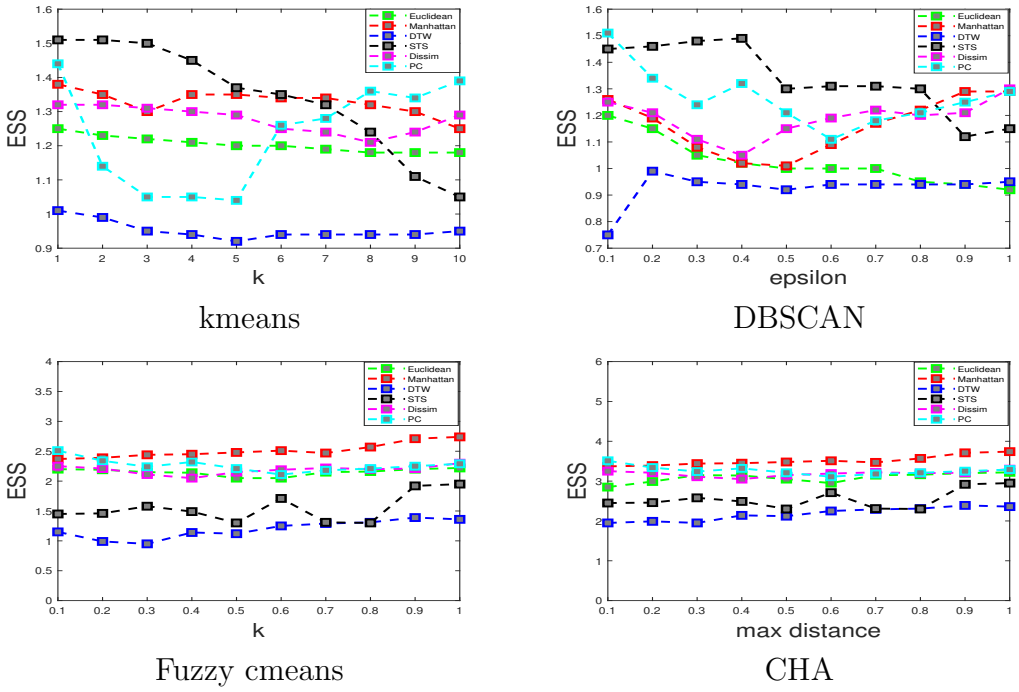


Figure 3: Quality of Returned Clusters on Urban Odense Traffic Data: ESS

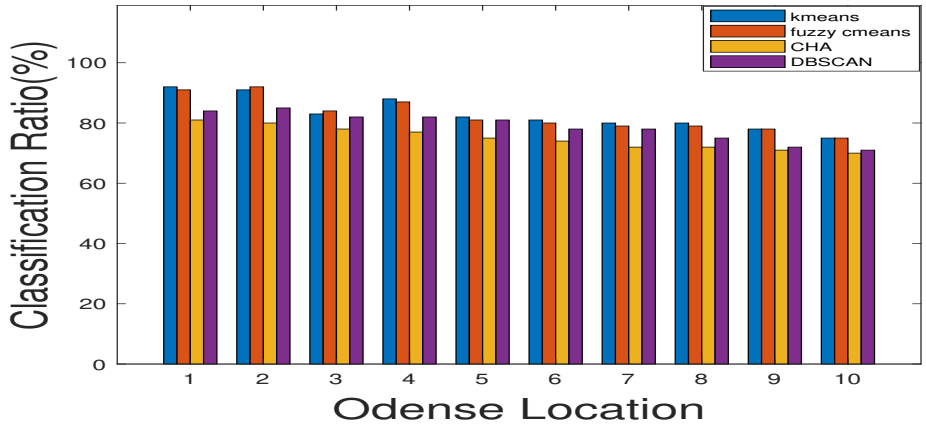


Figure 4: Quality of Returned Clusters on Urban Odense Traffic Data: Classification Ratio

The first experiments aim at comparing the space time series clustering using the large urban Odense traffic data. We have collected several time series in different locations. Each time series constitutes the set of the number of flow values in one hour, where each flow is the number of cars or bikes during a time period. We set the time period to 1min, 2min, 3min, 4min, and 5min, respectively. This allows to create different time series with different sizes (60, 30, 20, 15, 12). We have captured more than 1 million of time series for the experiments and constructed 20 datasets. Each dataset named $nXmY$ indicates that it contains n time series with m different flow values. In the experiments, n belongs to the set {1,000, 10,000, 100,000, 1,000,000}, and m belongs to the set {12, 15, 20, 30, 60}. Figures 3 and 4 showed the quality of the space time series clustering using ESS and classification ratio. Figure 3 showed the ESS value of the different space time series clustering algorithms on dataset $1,000,000X60Y$, along with different similarity measures (Euclidean, Manhattan, DTW, STS, Dissim, and PC). We have also observed that by varying the number of clusters in k-means and fuzzy c-means, respectively, from 1 to 10, epsilon and maximum distance in DBSCAN and hierarchical clustering, respectively from 0.1 to 1.0, DTW provides better results whatever the case used. This is explained by the fact that the DTW measure is well adopted for space time series data by considering both temporal and spatial dimensions of the space time series data. This is why the DTW measure is used for the remaining experiments. Figure 4 presents the classification ratio of different clustering algorithms, and with different Odense locations. The results revealed that the classification ratio of the space time series clustering algorithms is decreased while increasing the number of traffic flow values. Thus, for low density locations, the classification ratio of all algorithms exceeds 80% (k-means and fuzzy c-means reach 90%). However, for high density locations, the classification ratio goes under 70% for some algorithms such as DBSCAN and CHA. Figure 5, and 6 present the runtime in seconds and the memory usage in mega bytes for the space time series clustering using the 20 datasets of the urban Odense traffic data. By varying the number of space time series data from 1,000 to 1,000,000, and the the number of flow values from 12 to 60, we remark that CHA and c-means are slow. Actually, they require high computational resources to handle the large urban Odense traffic data. This confirms the discussion given in Section 3.4. In general, the space time series clustering algorithms need reasonable memory usage (less than 250 mb), but they require high computational resources (more than 3 hours) for handling 1,000,000 flow

values. We can state that the existing space time series clustering algorithms could handle the large data as the case of urban Odense traffic data.

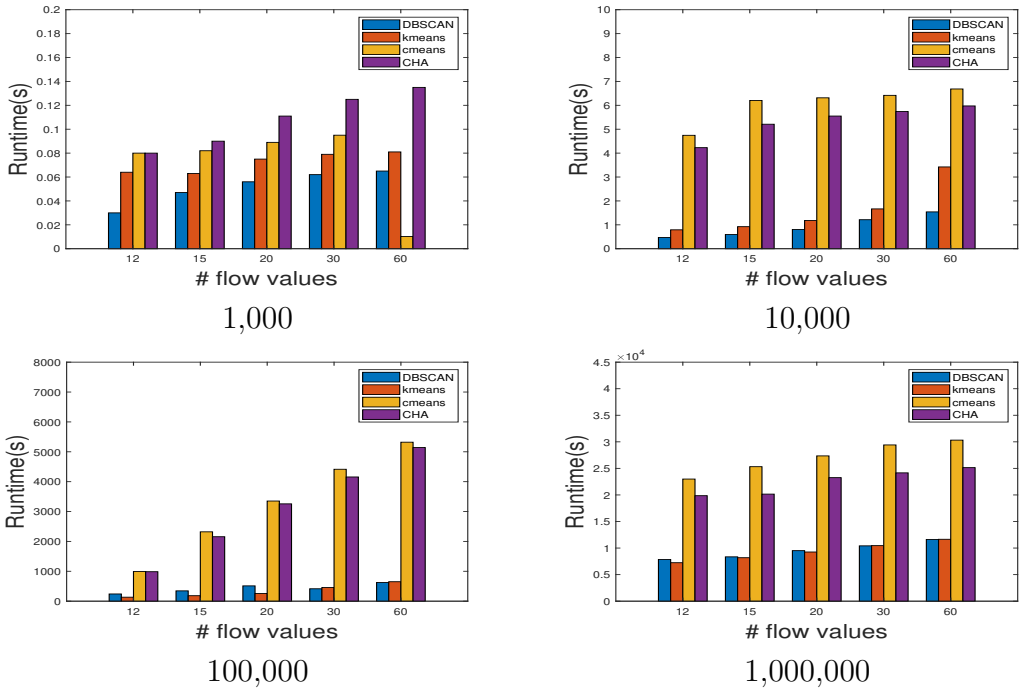


Figure 5: Runtime of Clustering on Urban Odense Traffic Data

4.6. Results on Urban Beijing Traffic Data

The next experiments aim to show the ability of the space time series clustering algorithms for handling big dataset, as the case of the urban traffic data captured in the second largest city in the world. Figure 7 presents the performance of the space time series clustering algorithms on Beijing locations. When varying the number of flow values from 1 million to 30 million, we have observed that the k-means and c-means provide good results in terms of quality of returned clusters for both ESS and classification ratio values. They also are very competitive in terms of runtime and memory usage compared to the other space time series clustering. However, all space time series clustering algorithms are high time consuming for dealing with big dataset; they need several days to group Beijing data having 30 million of flow values. More advanced clustering techniques Schubert et al. 2017;

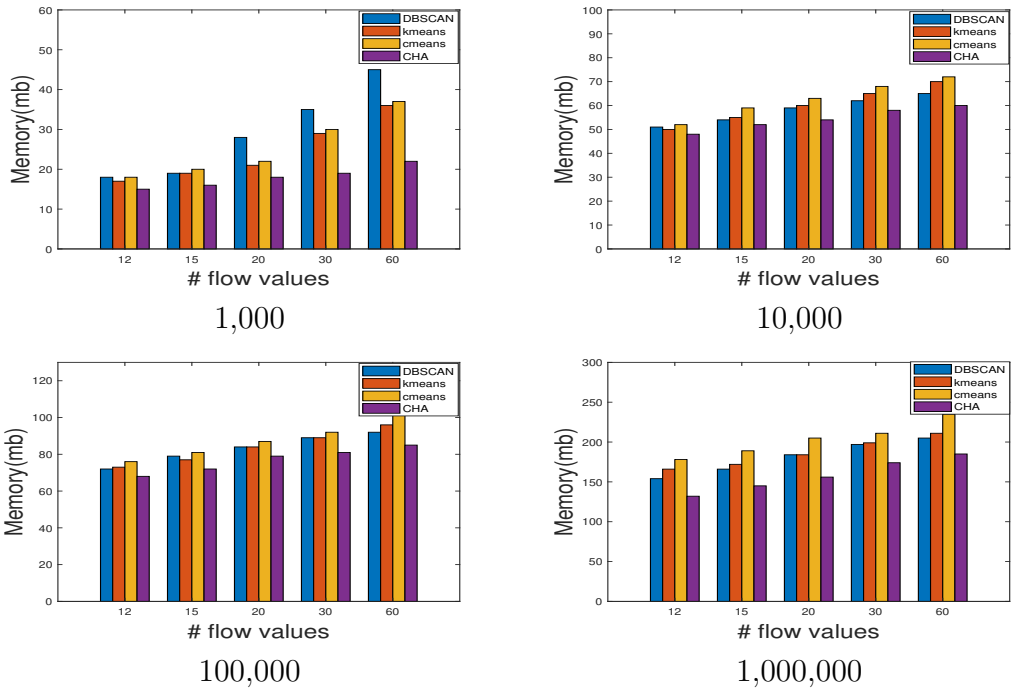


Figure 6: Memory of Clustering on Urban Odense Traffic Data

Pourkamali-Anaraki and Becker 2017; Q. Zhang et al. 2018; Xiaojun Chen et al. 2018 are needed to be adopted for handling big space time series datasets.

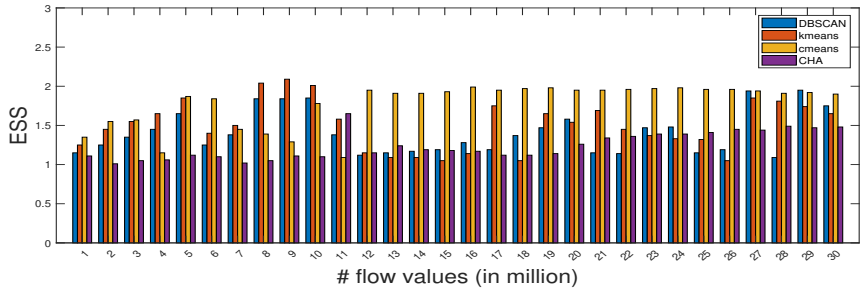
5. Challenges and Future Directions

This section presents some challenges and future applications in space time series clustering.

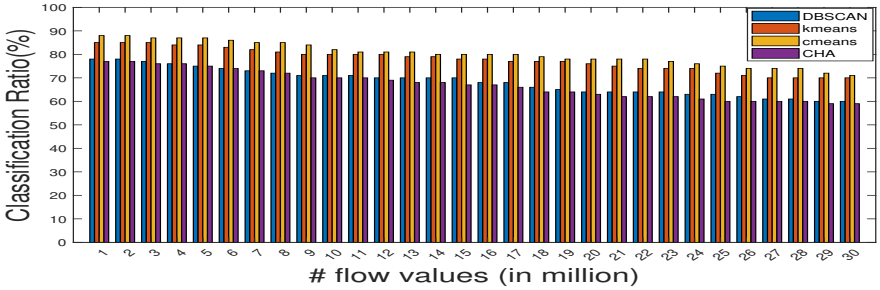
5.1. Challenges

In this section, we present four challenges in the future work on space time series clustering.

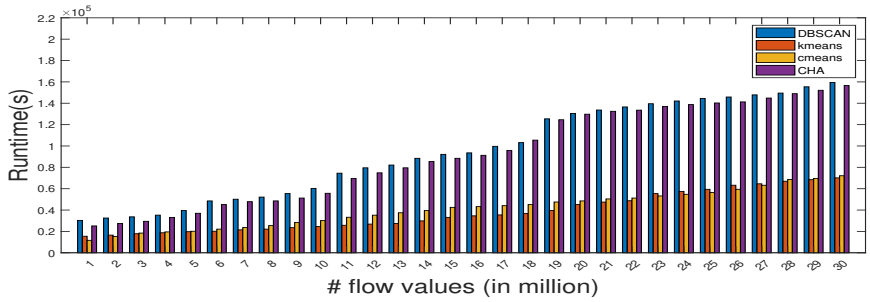
Challenge I: Improving runtime performance of space time series clustering. Space Time series clustering approaches are very time consuming in particular while dealing with many spatial points and huge time series. To handle the big space time series, technologies from different domains could be adapted such as: i) High performance computing (HPC) aims at using parallel frameworks to speed up the sequential solutions Shi et



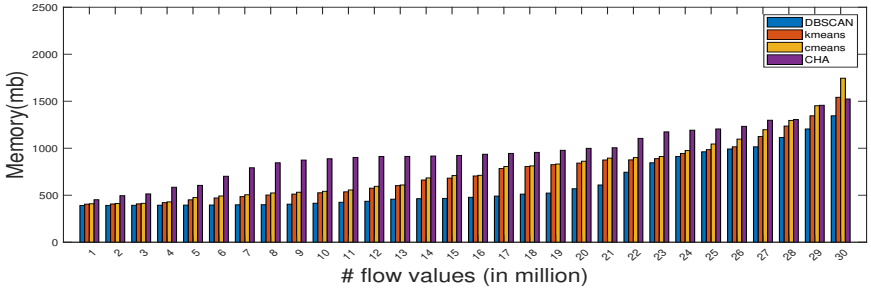
Quality: ESS



Quality: Classification Ratio



Runtime



Memory Consumption

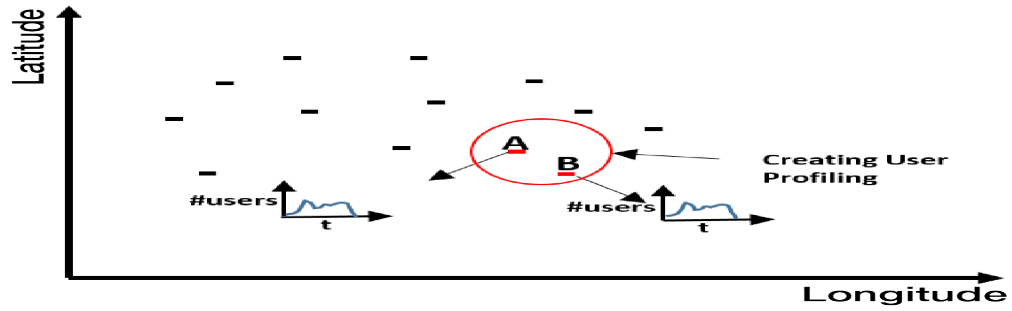
Figure 7: Results on Beijing Locations

al. 2018; Y. Djenouri, D. Djenouri, et al. 2019; Y. Djenouri, Bendjoudi, et al. 2014. Some of the most well-known architectures apply the multi-core CPU or GPU and perform on MapReduce or Spark platforms; ii) Computational intelligence (CI) is a collection of intelligent methods aiming at optimizing complex problems with strategies like meta-heuristics Queiroga, Subramanian, and Lucıdio dos Anjos 2018; Y. Djenouri, Drias, and Bendjoudi 2014; Y. Djenouri, Drias, and Habbas 2014; Y. Djenouri and Comuzzi 2017; and iii) Database systems provide techniques to efficiently store, update, and search space time series data, such as query optimization and index optimization. Adapting, combining, and optimizing technologies in space time series also provide many open research questions and future directions.

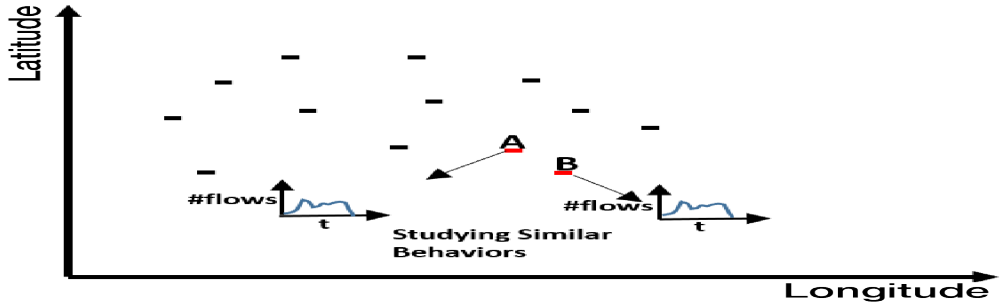
Challenge II: Improving quality performance of space time series clustering. The quality of the existing space time series clustering approaches became poor for the complex and big time series data. To solve this limitation, the deep network Ni et al. 2018 model can be utilized and applied for handling this situation. In this context, a suitable distribution of the data in the deep network should be considered and performed.

Challenge III: Correlation between space time series data. The existing algorithms for space time series clustering consider individual space time series data and ignore the correlation between the time series data. Studying the correlation between space time series using pattern mining algorithms Yagoubi et al. 2017; Campisano et al. 2018; Y. Djenouri, J. C.-W. Lin, et al. 2019; Y. Djenouri and Comuzzi 2017; Y. Djenouri, D. Djenouri, et al. 2019 could be helpful for space time series. Such complex or extended systems could be interesting for some excited applications such as urban traffic data Gonzalez et al. 2007.

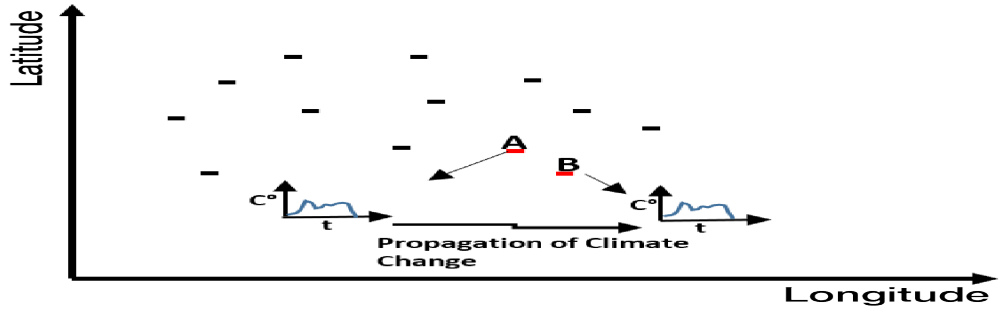
Challenge IV: Adaptation of advanced and specialized clustering methods Several variants of clustering models could also be adapted to handle the space time series data. Many adaptations to specific scenarios such as spatial data Ng and J. Han 2002; Birant and Kut 2007; J. C.-W. Lin, Y. Li, Fournier-Viger, Y. Djenouri, and J. Zhang 2019, high dimensional data McCallum, Nigam, and Ungar 2000; Rathore et al. 2018; Y. Djenouri, Bendjoudi, et al. 2015; J. C.-W. Lin, Y. Li, Fournier-Viger, Y. Djenouri, and L. S.-L. Wang 2019, time series data Keogh and J. Lin 2005, or streaming data Euán, Ombao, and Ortega 2018; McDowell et al. 2018 remain potential possibilities. All these special scenarios are somehow related to possible scenarios in tackling space time series data and thus studied methods in literature review for these scenarios could also be relevant for adaptations to



a. Social media analysis



b. Urban traffic flows



c. Climate change

Figure 8: Future directions in space time series clustering

space time series data and for tackling these different aspects simultaneously.

5.2. Future Directions

Myriad volumes of space time series data are collected in several applications and domains such as social media, urban traffic, and climate change. In this section, we briefly describe future directions and motivation for applying space time series clustering in different applications and domains.

5.2.1. Social Media Analysis

Social media analysis has received a great attention in World Wide Web. Finding structural properties in a social network such as Twitter is a challenge issue Kwon et al. 2013 in recent decades. Consider the example shown in Figure 8(a) that the tweets mentioned the crude oil prices profile[5]. At each location, we can observe that different number of users have accessed this page across several days for evaluation. Applying space time series clustering on these data can discover relevant information, for instance, the number of people located in red countries are approximately the same when the oil petrol changes a lot (highly increased or highly decreased). These could be explained by the fact that the economy of these countries are highly dependent of the oil prices, where people care for a high changes in the price of such natural resources. In general, applying space time series clustering in social media data allows creating user profiling in both spatial and temporal dimensions.

5.2.2. Urban Traffic Flows

Urban traffic data consists of observations like number and speed of cars or other vehicles at certain locations as measured by the deployed sensors. These numbers can be interpreted as traffic flow which in turn relates to the capacity of streets and the demand of the traffic system Belhadi, Y. Djenouri, and J. C.-W. Lin 2019; Y. Djenouri and Zimek 2018; Y. Djenouri, Zimek, and Chiarandini 2018. City planners are interested in studying the impact of various conditions on the traffic flow, leading to finding the correlation between the traffic flows. Consider the example shown in Figure 8(b), it illustrates urban traffic flow of different locations in a given city. For each location, we have observed different flow values represented by a time series. Applying space time series clustering on these data allows to group locations having similar traffic behaviors. For instance, traffic in red locations are quite similar. If the traffic flow increases in A, it increases in B as well.

5.2.3. Climate Change

Climate change directly effects on the precipitation all over the world. Several research Singh, Lo, and Qin 2017; Karpatne et al. 2013 focus on the changes in intensity and frequency of precipitation represented by a time

[5]https://twitter.com/CrudeOilPrices

series. Studying propagation of climate changes around closer locations is a challenge issue. One idea to solve this issue is to exploit space time series clustering. Consider the example shown in Figure 8(c), it presents the temperatures at different locations. At each location, we have observed different temperature values represented by a time series. Applying the space time series clustering on these data can allow us to group locations having similar climate change behaviors. For instance, temperature in red locations are quite similar and if the climate changes in A, it is also changed in B. In this case, we can say that there is a propagation in climate change between A and B.

6. Conclusion

This paper presented an overview on space time series clustering approaches. First, we have discussed three categories of existing clustering approaches, including hierarchical, partitioning, and overlapping space time series clustering. We have also elaborated on how limitations in one case could be beneficial in another case depending on the scenario and available knowledge. Second, we have explained four challenges of space time series clustering and discussed how they might affect the clustering process. Third, we have also provided a case study of existing space time series clustering algorithms on intelligent transportation, targeting to two smart cities (Odense in Denmark with large dataset and Beijing in China with big dataset). We have finally presented a summary of the most relevant directions that could be concluded from the applications of space time series clustering. Overall, whereas solutions to time series clustering has gained high maturity in domains such as image/speech processing, transportation, and bio-medical data; the use of space time series clustering in these domains has become an emerging issue. Our main conclusion from this study is that much exploration and deep progress are still required in all directions to obtain more mature solutions for end-user satisfaction.

References

Aghabozorgi, Saeed, Ali Seyed Shirkhorshidi, and Teh Ying Wah (2015). "Time-series clustering–A decade review". In: *Information Systems* 53, pp. 16–38.

Anand, Raj, Vishnu Pratap Singh Kirar, and Kavita Burse (2013). "K-fold cross validation and classification accuracy of pima Indian diabetes data set using higher order neural network and PCA". In: *Int. J. Soft Comput. Eng* 2.6, pp. 436–438.

Andrienko, Natalia and Gennady Andrienko (2013). "A visual analytics framework for spatio-temporal analysis and modelling". In: *Data Mining and Knowledge Discovery* 27.1, pp. 55–83.

Bai, Xue et al. (2014). "Mining high-temperature event space-time regions in geo-referenced temperature series data". In: *Fuzzy Systems and Knowledge Discovery (FSKD), 2014 11th International Conference on*. IEEE, pp. 671–676.

Barron, Andrew, Jorma Rissanen, and Bin Yu (1998). "The minimum description length principle in coding and modeling". In: *IEEE Transactions on Information Theory* 44.6, pp. 2743–2760.

Belhadi, Asma, Youcef Djenouri, and Jerry Chun-Wei Lin (2019). "Comparative Study on Trajectory Outlier Detection Algorithms". In: *2019 International Conference on Data Mining Workshops (ICDMW)*. IEEE, pp. 415–423.

Bezdek, James C, Robert Ehrlich, and William Full (1984). "FCM: The fuzzy c-means clustering algorithm". In: *Computers & Geosciences* 10.2-3, pp. 191–203.

Birant, Derya and Alp Kut (2007). "ST-DBSCAN: An algorithm for clustering spatial–temporal data". In: *Data & Knowledge Engineering* 60.1, pp. 208–221.

Campisano, Riccardo et al. (2018). "Discovering Tight Space-Time Sequences". In: *International Conference on Big Data Analytics and Knowledge Discovery*. Springer, pp. 247–257.

Chen, Xiaojun et al. (2018). "Purtreeclust: A clustering algorithm for customer segmentation from massive customer transaction data". In: *IEEE Transactions on Knowledge and Data Engineering* 30.3, pp. 559–572.

Cho, Myoungsu et al. (2014). "Stroscope: Multi-scale visualization of irregularly measured time-series data". In: *IEEE transactions on visualization and computer graphics* 20.5, pp. 808–821.

Cole, Richard, Dennis Shasha, and Xiaojian Zhao (2005). "Fast Window Correlations over Uncooperative Time Series". In: *Proceedings of the International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 743–749.

Dau, Hoang Anh, Nurjahan Begum, and Eamonn Keogh (2016). "Semi-supervision dramatically improves time series clustering under dynamic time warping". In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 999–1008.

Dave, Rajesh N and Kurra Bhaswan (1992). "Adaptive fuzzy c-shells clustering and detection of ellipses". In: *IEEE Transactions on Neural Networks* 3.5, pp. 643–662.

Deng, Min et al. (2018). "Heterogeneous space–time artificial neural networks for space–time series prediction". In: *Transactions in GIS* 22.1, pp. 183–201.

Di Lascio, F Marta L, Fabrizio Durante, and Roberta Pappada (2017). "Copula–based clustering methods". In: *Copulas and Dependence Models with Applications*. Springer, pp. 49–67.

Disegna, Marta, Pierpaolo D'Urso, and Fabrizio Durante (2017). "Copula-based fuzzy clustering of spatial time series". In: *Spatial Statistics* 21, pp. 209–225.

Djenouri, Youcef, Asma Belhadi, Jerry Chun-Wei Lin, and Alberto Cano (2019). "Adapted K-Nearest Neighbors for Detecting Anomalies on Spatio–Temporal Traffic Flow". In: *IEEE Access* 7, pp. 10015–10027.

Djenouri, Youcef, Asma Belhadi, Jerry Chun-Wei Lin, Djamel Djenouri, et al. (2019). "A Survey on Urban Traffic Anomalies Detection Algorithms". In: *IEEE Access* 7, pp. 12192–12205.

Djenouri, Youcef, Ahcene Bendjoudi, et al. (2014). "Parallel association rules mining using GPUS and bees behaviors". In: *2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR)*. IEEE, pp. 401–405.

— (2015). "GPU-based bees swarm optimization for association rules mining". In: *The Journal of Supercomputing* 71.4, pp. 1318–1344.

Djenouri, Youcef and Marco Comuzzi (2017). "Combining Apriori heuristic and bio-inspired algorithms for solving the frequent itemsets mining problem". In: *Information Sciences* 420, pp. 1–15.

Djenouri, Youcef, Djamel Djenouri, et al. (2019). "Exploiting GPU parallelism in improving bees swarm optimization for mining big transactional databases". In: *Information Sciences* 496, pp. 326–342.

Djenouri, Youcef, Habiba Drias, and Ahcene Bendjoudi (2014). "Pruning irrelevant association rules using knowledge mining". In: *International Journal of Business Intelligence and Data Mining* 9.2, pp. 112–144.

Djenouri, Youcef, Habiba Drias, and Zineb Habbas (2014). "Bees swarm optimisation using multiple strategies for association rule mining". In: *International Journal of Bio-Inspired Computation* 6.4, pp. 239–249.

Djenouri, Youcef, Jerry Chun-Wei Lin, et al. (2019). "Highly efficient pattern mining based on transaction decomposition". In: *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, pp. 1646–1649.

Djenouri, Youcef and Arthur Zimek (2018). "Outlier Detection in Urban Traffic Data". In: *Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics*, p. 3.

Djenouri, Youcef, Arthur Zimek, and Marco Chiarandini (2018). "Outlier detection in urban traffic flow distributions". In: *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, pp. 935–940.

Eftelioglu, Emre et al. (2016). "Ring-shaped hotspot detection". In: *IEEE Transactions on Knowledge and Data Engineering* 28.12, pp. 3367–3381.

Esling, Philippe and Carlos Agon (2012). "Time-series data mining". In: *ACM Computing Surveys (CSUR)* 45.1, p. 12.

Ester, Martin et al. (1996). "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Proceedings of KDD*, pp. 226–231.

Euán, Carolina, Hernando Ombao, and Joaquın Ortega (2018). "The hierarchical spectral merger algorithm: a new time series clustering procedure". In: *Journal of Classification* 35.1, pp. 71–99.

Feng, Zhenni and Yanmin Zhu (2016). "A survey on trajectory data mining: techniques and applications". In: *IEEE Access* 4, pp. 2056–2067.

Ferstl, Florian et al. (2017). "Time-Hierarchical Clustering and Visualization of Weather Forecast Ensembles". In: *IEEE transactions on visualization and computer graphics* 23.1, pp. 831–840.

Fournier-Viger, Philippe et al. (2014). "SPMF: a Java open-source pattern mining library". In: *The Journal of Machine Learning Research* 15.1, pp. 3389–3393.

Frentzos, Elias, Kostas Gratsias, and Yannis Theodoridis (2007). "Index-based most similar trajectory search". In: *2007 IEEE 23rd International Conference on Data Engineering*. IEEE, pp. 816–825.

Fu, Tak-chung (2011). "A review on time series data mining". In: *Engineering Applications of Artificial Intelligence* 24.1, pp. 164–181.

Gharavi, H and B Hu (2017). "Space-Time Approach for Disturbance Detection and Classification". In: *IEEE Transactions on Smart Grid*.

Gholami, Behnam and Vladimir Pavlovic (2017). "Probabilistic Temporal Subspace Clustering". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3066–3075.

Golay, Xavier et al. (1998). "A new correlation-based fuzzy logic clustering algorithm for FMRI". In: *Magnetic Resonance in Medicine* 40.2, pp. 249–260.

Gonzalez, Hector et al. (2007). "Adaptive fastest path computation on a road network: a traffic mining approach". In: *Proceedings of the 33rd international conference on Very large data bases*, pp. 794–805.

Guha, Sudipto, Rajeev Rastogi, and Kyuseok Shim (1998). "CURE: an efficient clustering algorithm for large databases". In: *ACM Sigmod Record*. Vol. 27. 2, pp. 73–84.

— (2000). "ROCK: A robust clustering algorithm for categorical attributes". In: *Information systems* 25.5, pp. 345–366.

Gupta, Manish et al. (2014). "Outlier detection for temporal data: A survey". In: *IEEE Transactions on Knowledge and Data Engineering* 26.9, pp. 2250–2267.

Hallac, David et al. (2017). "Toeplitz inverse covariance-based clustering of multivariate time series data". In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 215–223.

Heredia, Luis Carlos Castro and Armando Rodrigo Mor (2019). "Density-based clustering methods for unsupervised separation of partial discharge sources". In: *International Journal of Electrical Power & Energy Systems* 107, pp. 224–230.

Hoeffding, Wassily (1963). "Probability inequalities for sums of bounded random variables". In: *Journal of the American statistical association* 58.301, pp. 13–30.

Hu, Jilin et al. (2018). "Risk-aware path selection with time-varying, uncertain travel costs: a time series approach". In: *The VLDB Journal* 27.2, pp. 179–200.

Izakian, Hesam and Witold Pedrycz (2014a). "Agreement-based fuzzy C-means for clustering data with blocks of features". In: *Neurocomputing* 127, pp. 266–280.

— (2014b). "Anomaly detection and characterization in spatial time series data: A cluster-centric approach". In: *IEEE Transactions on Fuzzy Systems* 22.6, pp. 1612–1624.

Izakian, Hesam, Witold Pedrycz, and Iqbal Jamal (2013). "Clustering spatiotemporal data: An augmented fuzzy c-means". In: *IEEE transactions on fuzzy systems* 21.5, pp. 855–868.

— (2015). "Fuzzy clustering of time series data using dynamic time warping distance". In: *Engineering Applications of Artificial Intelligence* 39, pp. 235–244.

Jain, Anil K, M Narasimha Murty, and Patrick J Flynn (1999). "Data clustering: a review". In: *ACM Computing Surveys (CSUR)* 31.3, pp. 264–323.

Jensen, Morten Bornø et al. (2016). "Vision for looking at traffic lights: Issues, survey, and perspectives". In: *IEEE Transactions on Intelligent Transportation Systems* 17.7, pp. 1800–1815.

Jiang, Jianhua, Yujun Chen, et al. (2019). "DPC-LG: Density peaks clustering based on logistic distribution and gravitation". In: *Physica A: Statistical Mechanics and its Applications* 514, pp. 25–35.

Jiang, Jianhua, Dehao Hao, et al. (2018). "GDPC: Gravitation-based density peaks clustering algorithm". In: *Physica A: Statistical Mechanics and its Applications* 502, pp. 345–355.

Jiang, Xiaoping, Chenghua Li, and Jing Sun (2018). "A modified K-means clustering for mining of multimedia databases based on dimensionality reduction and similarity measures". In: *Cluster Computing* 21.1, pp. 797–804.

Karpatne, Anuj et al. (2013). "Earth science applications of sensor data". In: *Managing and Mining Sensor Data*. Springer, pp. 505–530.

Karypis, George (2002). *CLUTO-a clustering toolkit*. Tech. rep. MINNESOTA UNIV MINNEAPOLIS DEPT OF COMPUTER SCIENCE.

Karypis, George, Eui-Hong Han, and Vipin Kumar (1999). "Chameleon: Hierarchical clustering using dynamic modeling". In: *Computer* 32.8, pp. 68–75.

Kaufman, Leonard and Peter J Rousseeuw (1990). "Partitioning around medoids (program pam)". In: *Finding groups in data: an introduction to cluster analysis*, pp. 68–125.

— (2009). *Finding groups in data: an introduction to cluster analysis*. Vol. 344. John Wiley & Sons.

Keogh, Eamonn and Shruti Kasetty (2003). "On the need for time series data mining benchmarks: a survey and empirical demonstration". In: *Data Mining and knowledge discovery* 7.4, pp. 349–371.

Keogh, Eamonn and Jessica Lin (2005). "Clustering of time-series subsequences is meaningless: implications for previous and future research". In: *Knowledge and information systems* 8.2, pp. 154–177.

Koperski, Krzysztof, Junas Adhikary, and Jiawei Han (1996). "Spatial data mining: progress and challenges survey paper". In: *ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pp. 1–10.

Kriegel, Hans-Peter, Peer Kröger, and Arthur Zimek (2009). "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 3.1, p. 1.

Krüger, Björn et al. (2017). "Efficient unsupervised temporal segmentation of motion data". In: *IEEE Transactions on Multimedia* 19.4, pp. 797–812.

Kwon, Sejeong et al. (2013). "Prominent features of rumor propagation in online social media". In: *2013 IEEE 13th International Conference on Data Mining*, pp. 1103–1108.

Li, Hailin (2019). "Multivariate time series clustering based on common principal component analysis". In: *Neurocomputing* 349, pp. 239–247.

Li, Li et al. (2015). "Trend modeling for traffic time series analysis: An integrated study". In: *IEEE Transactions on Intelligent Transportation Systems* 16.6, pp. 3430–3439.

Liao, T Warren (2005). "Clustering of time series data-a survey". In: *Pattern recognition* 38.11, pp. 1857–1874.

Lin, Jerry Chun-Wei, Yuanfa Li, Philippe Fournier-Viger, Youcef Djenouri, and Leon Shyue-Liang Wang (2019). "Mining High-Utility Sequential Patterns from Big Datasets". In: *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, pp. 2674–2680.

Lin, Jerry Chun-Wei, Yuanfa Li, Philippe Fournier-Viger, Youcef Djenouri, and Ji Zhang (2019). "An Efficient Chain Structure to Mine High-Utility Sequential Patterns". In: *2019 International Conference on Data Mining Workshops (ICDMW)*. IEEE, pp. 1013–1019.

Lin, Jessica, Eamonn Keogh, Stefano Lonardi, et al. (2003). "A symbolic representation of time series, with implications for streaming algorithms". In: *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*. ACM, pp. 2–11.

Lin, Jessica, Eamonn Keogh, Li Wei, et al. (2007). "Experiencing SAX: a novel symbolic representation of time series". In: *Data Mining and knowledge discovery* 15.2, pp. 107–144.

Liu, Yan (2015). "Scalable Multivariate Time-Series Models for Climate Informatics". In: *Computing in Science & Engineering* 17.6, pp. 19–26.

MacQueen, James et al. (1967). "Some methods for classification and analysis of multivariate observations". In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA, pp. 281–297.

McCallum, Andrew, Kamal Nigam, and Lyle H Ungar (2000). "Efficient clustering of high-dimensional data sets with application to reference matching". In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 169–178.

McDowell, Ian C et al. (2018). "Clustering gene expression time series data using an infinite Gaussian process mixture model". In: *PLoS computational biology* 14.1, e1005896.

Mikalsen, Karl Øyvind et al. (2018). "Time series cluster kernel for learning similarities between multivariate time series with missing data". In: *Pattern Recognition* 76, pp. 569–581.

Möller-Levet, Carla S et al. (2003). "Fuzzy clustering of short time-series and unevenly distributed sampling points". In: *International Symposium on Intelligent Data Analysis*. Springer, pp. 330–340.

Morales-Esteban, Antonio et al. (2014). "A fast partitioning algorithm using adaptive Mahalanobis clustering with application to seismic zoning". In: *Computers & Geosciences* 73, pp. 132–141. ISSN: 0098-3004.

Mori, Usue, Alexander Mendiburu, and Jose A Lozano (2016). "Distance measures for time series in R: The TSdist package". In: *R journal* 8.2, pp. 451–459.

Nanopoulos, Alexandros, Yannis Theodoridis, and Yannis Manolopoulos (2001). "C2P: clustering based on closest pairs". In: *VLDB*, pp. 331–340.

Ng, Raymond T. and Jiawei Han (2002). "CLARANS: A method for clustering objects for spatial data mining". In: *IEEE transactions on knowledge and data engineering* 14.5, pp. 1003–1016.

Ni, Jingchao et al. (2018). "Co-Regularized Deep Multi-Network Embedding". In: *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pp. 469–478.

Paci, Lucia and Francesco Finazzi (2017). "Dynamic model-based clustering for spatio-temporal data". In: *Statistics and Computing*, pp. 1–16.

Penfold, Robert B et al. (2016). "Space–Time Cluster Analysis to Detect Innovative Clinical Practices: A Case Study of Aripiprazole in the Department of Veterans Affairs". In: *Health services research*.

Pourkamali-Anaraki, Farhad and Stephen Becker (2017). "Preconditioned data sparsification for big data with applications to PCA and K-means". In: *IEEE Transactions on Information Theory* 63.5, pp. 2954–2974.

Putri, Givanna H et al. (2019). "ChronoClust: Density-based clustering and cluster tracking in high-dimensional time-series data". In: *Knowledge-Based Systems* 174, pp. 9–26.

Qu, Licheng et al. (2019). "Daily long-term traffic flow forecasting based on a deep neural network". In: *Expert Systems with Applications* 121, pp. 304–312.

Queiroga, Eduardo, Anand Subramanian, and F Cabral Lucıdio dos Anjos (2018). "Continuous greedy randomized adaptive search procedure for data clustering". In: *Applied Soft Computing* 72, pp. 43–55.

Rathore, Punit et al. (2018). "A Rapid Hybrid Clustering Algorithm for Large Volumes of High Dimensional Data". In: *IEEE Transactions on Knowledge and Data Engineering*.

Rezaei, Shahbaz and Xin Liu (2019). "Deep learning for encrypted traffic classification: An overview". In: *IEEE communications magazine* 57.5, pp. 76–81.

Rodrigues, Pedro Pereira, João Gama, and Joao Pedroso (2008). "Hierarchical clustering of time-series data streams". In: *IEEE transactions on knowledge and data engineering* 20.5, pp. 615–627.

Rodriguez, Alex and Alessandro Laio (2014). "Clustering by fast search and find of density peaks". In: *Science* 344.6191, pp. 1492–1496.

Sakoe, Hiroaki and Seibi Chiba (1978). "Dynamic programming algorithm optimization for spoken word recognition". In: *IEEE transactions on acoustics, speech, and signal processing* 26.1, pp. 43–49.

Schubert, Erich et al. (2017). "DBSCAN revisited, revisited: why and how you should (still) use DBSCAN". In: *ACM Transactions on Database Systems (TODS)* 42.3, p. 19.

Shekhar, Shashi, Michael R Evans, et al. (2011). "Identifying patterns in spatial information: A survey of methods". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 1.3, pp. 193–214.

Shekhar, Shashi, Zhe Jiang, et al. (2015). "Spatiotemporal data mining: a computational perspective". In: *ISPRS International Journal of Geo-Information* 4.4, pp. 2306–2338.

Shen, Jianan and Tao Cheng (2016). "A framework for identifying activity groups from individual space-time profiles". In: *International Journal of Geographical Information Science* 30.9, pp. 1785–1805.

Shi, Xuanhua et al. (2018). "Graph processing on GPUs: A survey". In: *ACM Computing Surveys (CSUR)* 50.6, p. 81.

Shieh, Jin and Eamonn Keogh (2008). "iSAX: Indexing and Mining Terabyte Sized Time Series". In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 623–631.

Singh, Saurabh K, Edmond Yat-Man Lo, and Xiaosheng Qin (2017). "Cluster Analysis of Monthly Precipitation over the Western Maritime Continent under Climate Change". In: *Climate* 5.4, p. 84.

Steiger, Enrico, Bernd Resch, João Porto de Albuquerque, et al. (2016). "Mining and correlating traffic events from human sensor observations with official transport data using self-organizing-maps". In: *Transportation Research Part C: Emerging Technologies* 73, pp. 91–104.

Steiger, Enrico, Bernd Resch, and Alexander Zipf (2016). "Exploration of spatiotemporal and semantic clusters of Twitter data using unsupervised neural networks". In: *International Journal of Geographical Information Science* 30.9, pp. 1694–1716.

Sumit, Sakhawat Hosain and Shamim Akhter (2019). "C-means clustering and deep-neuro-fuzzy classification for road weight measurement in traffic management system". In: *Soft Computing* 23.12, pp. 4329–4340.

Sun, Tao et al. (2017). "Degree-Pruning Dynamic Programming Approaches to Central Time Series Minimizing Dynamic Time Warping Distance". In: *IEEE transactions on cybernetics* 47.7, pp. 1719–1729.

Vesanto, Juha and Esa Alhoniemi (2000). "Clustering of the self-organizing map". In: *IEEE Transactions on neural networks* 11.3, pp. 586–600.

Von Landesberger, Tatiana et al. (2016). "Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering". In: *IEEE transactions on visualization and computer graphics* 22.1, pp. 11–20.

Wang, Lei et al. (2016). "Group behavior time series anomaly detection in specific network space based on separation degree". In: *Cluster Computing* 19.3, pp. 1201–1210.

Wang, Xiao et al. (2019). "Clustering of interval-valued time series of unequal length based on improved dynamic time warping". In: *Expert Systems with Applications* 125, pp. 293–304.

Wang, Yi et al. (2016). "Clustering of electricity consumption behavior dynamics toward big data applications". In: *IEEE transactions on smart grid* 7.5, pp. 2437–2447.

Wu, Yingcai et al. (2017). "StreamExplorer: A Multi-Stage System for Visually Exploring Events in Social Streams". In: *IEEE Transactions on Visualization and Computer Graphics*.

Xi, Rui et al. (2018). "Deep Dilation on Multimodality Time Series for Human Activity Recognition". In: *IEEE Access* 6, pp. 53381–53396.

Xiong, Hui, Junjie Wu, and Jian Chen (2009). "K-means clustering versus validation measures: a data-distribution perspective". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 39.2, pp. 318–331.

Xiong, Yimin and Dit-Yan Yeung (2002). "Mixtures of ARMA models for model-based time series clustering". In: *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pp. 717–720.

Yagoubi, Djamel Edine et al. (2017). "DPiSAX: Massively distributed partitioned iSAX". In: *2017 IEEE International Conference on Data Mining (ICDM)*, pp. 1135–1140.

Yi, Byoung-Kee and Christos Faloutsos (2000). "Fast time sequence indexing for arbitrary Lp norms". In: *VLDB*. Vol. 385. 394. Citeseer, p. 99.

Yu, Hwa-Lung et al. (2015). "Analysis of space–time patterns of rainfall events during 1996–2008 in Yilan County (Taiwan)". In: *Stochastic environmental research and risk assessment* 29.3, pp. 929–945.

Zhang, Qingchen et al. (2018). "High-order possibilistic c-means algorithms based on tensor decompositions for big data in IoT". In: *Information Fusion* 39, pp. 72–80.

Zhang, Tian, Raghu Ramakrishnan, and Miron Livny (1996). "BIRCH: an efficient data clustering method for very large databases". In: *ACM Sigmod Record*. Vol. 25. 2, pp. 103–114.

Zhang, Yongli et al. (2017). ""Serial" versus "Parallel": A Comparison of Spatio-Temporal Clustering Approaches". In: *International Symposium on Methodologies for Intelligent Systems*. Springer, pp. 396–403.

Zheng, Yu (2015). "Trajectory data mining: an overview". In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 6.3, p. 29.

Zhou, Xun, Shashi Shekhar, and Reem Y Ali (2014). "Spatiotemporal change footprint pattern discovery: an inter-disciplinary survey". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 4.1, pp. 1–23.

Zolhavarieh, Seyedjamal, Saeed Aghabozorgi, and Ying Wah Teh (2014). "A review of subsequence time series clustering". In: *The Scientific World Journal* 2014.