Chapter

Clustering of Time-Series Data

Esma Ergüner Özkoç

Abstract

The process of separating groups according to similarities of data is called "clustering." There are two basic principles: (i) the similarity is the highest within a cluster and (ii) similarity between the clusters is the least. Time-series data are unlabeled data obtained from different periods of a process or from more than one process. These data can be gathered from many different areas that include engineering, science, business, finance, health care, government, and so on. Given the unlabeled time-series data, it usually results in the grouping of the series with similar characteristics. Time-series clustering methods are examined in three main sections: data representation, similarity measure, and clustering algorithm. The scope of this chapter includes the taxonomy of time-series data clustering and the clustering of gene expression data as a case study.

Keywords: time-series data, data mining, data representation, similarity measure, clustering algorithms, gene expression data clustering

1. Introduction

The rapid development of technology has led to the registration of many processes in an electronic environment, the storage of these records, and the accessibility of these records when requested. With the evolving technology such as cloud computing, big data, the accumulation of a large amount of data stored in databases, and the process of parsing and screening useful information made data mining necessary.

It is possible to examine the data which are kept in databases and reach to huge amounts of size every second, in two parts according to their changes in time: static and temporal. Data is called the static data when its feature values do not change with time, if the feature comprise values change with time then it is called the temporal or time-series data.

Today, with the increase in processor speed and the development of storage technologies, real-world applications can easily record changing data over time.

Time-series analysis is a trend study subject because of its prevalence in various fields ranging from science, engineering, bioinformatics, finance, and government to health-care applications [1–3]. Data analysts are looking for the answers of such questions: Why does the data change this way? Are there any patterns? Which series show similar patterns? etc. Subsequence matching, indexing, anomaly detection, motif discovery, and clustering of the data are the answers of some questions [4]. Clustering, which is one of the most important concepts of data mining, defines its structure by separating unlabeled data sets into homogeneous groups. Many general-purpose clustering algorithms are used for the clustering of time-series data, either by directly or by evolving. Algorithm selection depends entirely on the purpose of the application and on the properties of the data such as sales data, exchange rates in finance, gene expression data, image data for face recognition, etc.

In the age of informatics, the analysis of multidimensional data that has emerged as part of the digital transformation in every field has gained considerable importance. These data can be from data received at different times from one or more sensors, stock data, or call records to a call center. This type of data, that is, observing the movement of a variable over time, where the results of the observation are distributed according to time, is called time-series data. Time-series analysis is used for many purposes such as future forecasts, anomaly detection, subsequence matching, clustering, motif discovery, indexing, etc. Within the scope of this study, the methods developed for the time-series data clustering which are important for every field of digital life in three main sections. In the first section, the proposed methods for the preparation of multidimensional data for clustering (dimension reduction) in the literature are categorized. In the second section, the similarity criteria to be used when deciding on the objects to be assigned to the related cluster are classified. In the third section, clustering algorithms of time-series data are examined under five main headings according to the method used. In the last part of the study, the use of time-series clustering in bioinformatics which is one of the favorite areas is included.

2. Time-series clustering approaches

There are many different categorizations of time-series clustering approaches. Such as, time-series clustering approaches can be examined in three main sections according to the characteristics of the data used whether they process directly on raw data, indirectly with features extracted from the raw data, or indirectly with models built from the raw data [5]. Another category is according to the clustering method: shape-based, feature-based, and model-based [6]. But whatever the categorization is, for any time-series clustering approach, the main points to be considered are: how to measure the similarity between time series; how to compress the series or reduce dimension and what algorithm to use for cluster. Therefore, this chapter examines time-series clustering approaches according to three main building blocks: data representation methods, distance measurements, and clustering algorithms (**Figure 1**).

2.1 Data representation

Data representation is one of the main challenging issues for time-series clustering. Because, time-series data are much larger than memory size [7, 8] that increases the need for high processor power and time for the clustering process increases exponentially. In addition, the time-series data are multidimensional, which is a difficulty for many clustering algorithms to handle, and it slows down the calculation of the similarity measurement. Consequently, it is very important for time-series data to represent the data without slowing down the algorithm execution time and without a significant data loss. Therefore, some requirements can be listed for any data representation methods [9]:

i. Significantly reduce the data size/dimensionality,

ii. Maintain the local and global shape characteristics of the time series,

iii. Acceptable computational cost,

iv. Reasonable level of reconstruction from the reduced representation,

v. Insensitivity to noise or implicit noise handling.

Dimension reduction is one of the most frequently used methods in the literature [7, 10–12] for the data representation.

Definition:

The representation of a time series T with length n is a model $\bar{T}$ with reduced dimensions, so that T approximates T [13]. Dimension reduction or feature extraction is a very useful method for reducing the number of variables/attributes or units

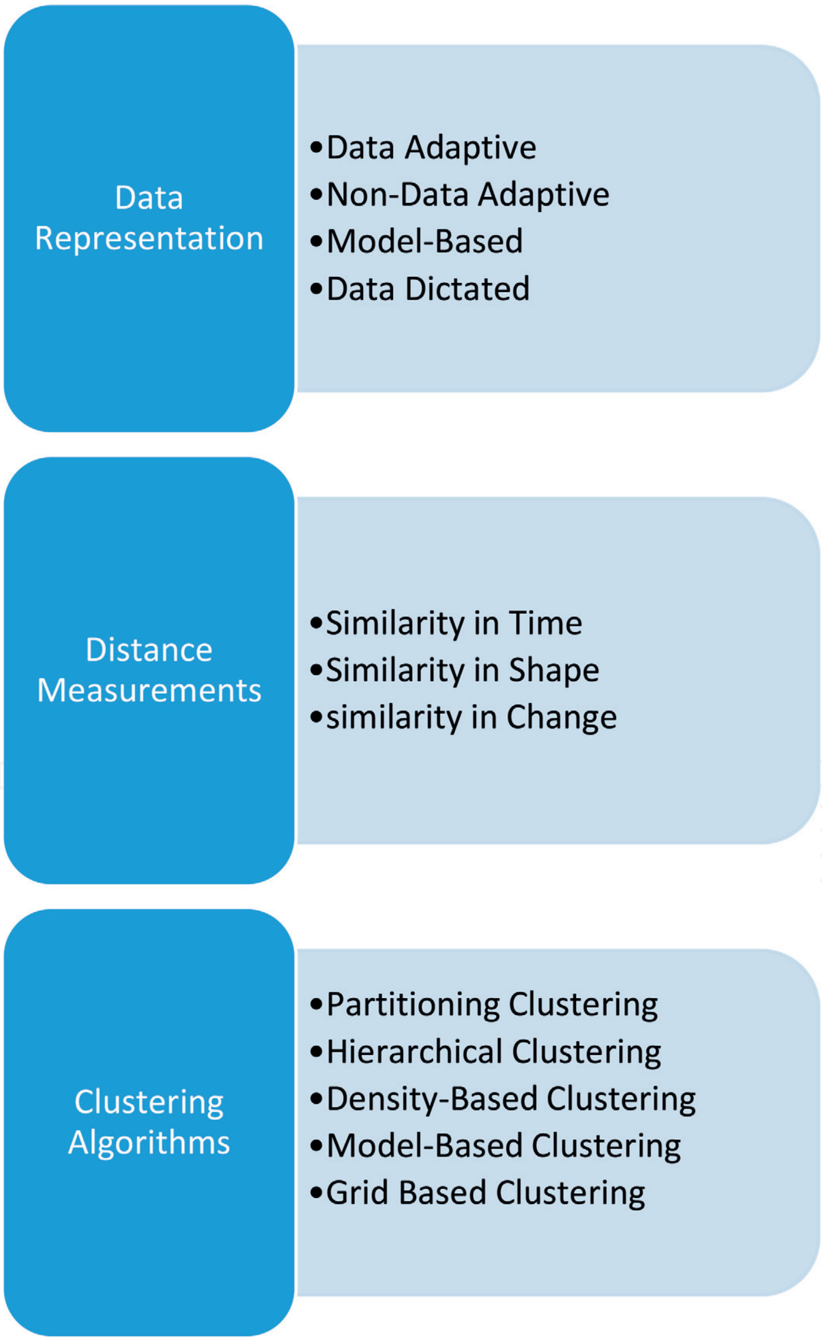

Figure 1.
Time-series clustering.

in multivariate statistical analyzes so that the number of attributes can be reduced to a number that "can handle."

Due to the noisy and high-dimensional features of many time-series data, data representations have been studied and generally examined in four main sections: data adaptive, nondata adaptive, model-based, and data dictated [6].

- **Data adaptive methods** that have changing parameters according to processing time-series data. Methods in this category try to minimize global reconstruction error by using unequal length segments. Although it is difficult to compare several time series, this method approximates each series better. Some of the popular data adaptive representation methods are: Symbolic Aggregate Approximation (SAX) [14], Adaptive Piecewise Constant Approximation (APCA) [15], Piecewise Linear Approximation (PLA) [16], Singular Value Decomposition (SVD) [17, 18], and Symbolic Natural Language (NLG) [19].

- **Non-data adaptive methods** are use fix-size parameters for the representing time-series data. Following methods are shown among non-data adaptive representation methods: Discrete Fourier Transform (DFT) [18], Discrete Wavelet Transform (DWT) [20–22], Discrete Cosine Transformation (DCT) [17], Perceptually Important Point (PIP) [23], Piecewise Aggregate Approximation (PAA) [24], Chebyshev Polynomials (CHEB) [25], Random Mapping [26], and Indexable Piecewise Linear Approximation (IPLA) [27].

- **Model-based methods** assume that observed time series was produced by an underlying model. The real issue here is to find the parameters that produce this model. Two time series produced by the same set of parameters using the underlying model are considered similar. Some of the model-based methods can be listed as: Auto-regressive Moving Average (ARMA) [28, 29], Time-Series Bitmaps [30], and Hidden Markov Model (HMM) [31–33].

- **Data dictated methods** automatically determine the dimension reduction rate but in the three methods mentioned above, the dimension reduction rates are automatically determined by the user. The most common example of data dictated method is clipped data [34–36].

Many representation methods for time-series data are proposed and each of them offering different trade-offs between the aforementioned requirements. The correct selection of the representation method plays a major role in the effectiveness and usability of the application to be performed.

2.2 Similarity/distance measure

In particular, the similarity measure is the most essential ingredient of time-series clustering.

The similarity or distance for the time-series clustering is approximately calculated, not based on the exact match as in traditional clustering methods. It requires to use distance function to compare two time series. In other words, the similarity of the time series is not calculated, it is estimated. If the estimated distance is large, the similarity between the time series is less and vice versa.

Definition:

Similarity between two "n" sized time series T = {$t_1,t_2,....t_n$} and U = {$u_1,u_2,....u_n$} is the length of the path connecting pair of points [11]. This distance is the measure of similarity. D (T, U) is a function that takes two times series (T, U) as input and calculates their distance "d".

Metrics to be used in clustering must cope with the problems caused by common features of time-series data such as noise, temporal drift, longitudinal scaling, offset translation, linear drift, discontinuities, and amplitude scaling. Various methods have been developed for similarity measure, and the method to choose is problem specific. These methods can be grouped under three main headings: similarity in time, similarity in shape, and similarity in change.

2.2.1 Similarity in time

The similarity between the series is that they are highly time dependent. Such a measure is costly for the raw time series, so a preprocessing or transformation is required beforehand [34, 36].

2.2.2 Similarity in shape

Clustering algorithms that use similarity in shape measure, assigns time series containing similar patterns to the same cluster. Independently of the time, it does not care how many times the pattern exists [37, 38].

2.2.3 Similarity in change

The result of using this metric is time-series clusters that have the similar autocorrelation structure. Besides, it is not a suitable metric for short time series [29, 39, 40].

2.3 Clustering algorithms

The process of separating groups according to similarities of data is called "clustering." There are two basic principles: the similarity within the cluster is the highest and the similarity between the clusters is the least. Clustering is done on the basis of the characteristics of the data and using multivariate statistical methods. When dividing data into clusters, the similarities/distances of the data to each other are measured according to the specification of the data (discrete, continuous, nominal, ordinal, etc.)

Han and Kamber [41] classify the general-purpose clustering algorithms which are actually designed for static data in five main sections: partition-based, hierarchical-based, density-based, grid-based, and model-based. Besides these, a wide variety of algorithms has been developed for time-series data. However, some of these algorithms (ignore minor differences) intend to directly use the methods developed for static data without changing the algorithm by transforming it into a static data form from temporal data. Some approaches apply a preprocessing step on the data to be clustered before using the clustering algorithm. This preprocessing step converts the raw-time-series data into feature vectors using dimension reduction techniques, or converts them into parameters of a specified model [42].

Definition:

Given a dataset on n time series T = {t_1, t_2,...., t_n}, time-series clustering is the process of partitioning of T into C = {C_1,C_2,....,C_k} according to certain similarity criterion. C_i is called "cluster" where,

$$T = \bigcup_{i=1}^{k} C_i \text{ and } C_i \bigcap C_j = \varnothing \text{ for i} \neq \text{j} \tag{1}$$

In this section, previously developed clustering algorithms will be categorized. Some of these algorithms work directly with raw time-series data, while others use the data presentation techniques that are previously mentioned.

Clustering algorithms are generally classified as: partitioning, hierarchical, graph-based, model-based, and density-based clustering.

2.3.1 Partitioning clustering

The K-means [43] algorithm is a typical partition-based clustering algorithm such that the data are divided into a number of predefined sets by optimizing the predefined criteria. The most important advantage is its simplicity and speed. So it can be applied to large data sets. However, the algorithm may not produce the same result in each run and cannot handle the outlier. Self-organizing map [44] is stronger than the noisy data clustering from K-means. The user is prompted to enter the cluster number and grid sets. It is difficult to determine the number of clusters for time-series data. Other examples of partition-based clustering are CLARANS [45] and K-medoids [46]. In addition, the partitioning approach is suitable for low-dimensional, well-separated data. However, time-series data are multidimensional and often contain intersections, embedded clusters.

In essence, these algorithms act as n-dimensional vectors to time-series data and applies distance or correlation functions to determine the amount of similarity between two series. Euclidean distance, Manhattan distance, and Pearson correlation coefficient are the most commonly used functions.

2.3.2 Hierarchical clustering

Contrary to the partitioning approach, which aims segmenting data that do not intersect, the hierarchical approach produces a hierarchical series of nested clusters that can be represented graphically (dendrogram, tree-like diagram). The branches of the dendrogram show the similarity between the clusters as well as the knowledge of the shaping of the clusters. Determined number of clusters can be obtained by cutting the dendrogram at a certain level.

Hierarchical clustering methods [47–49] are based on the separating clusters into subgroups that are processed step by step as a whole, or the stepwise integration of individual clusters into a cluster [50]. Hierarchical clustering methods are divided into two methods: agglomerative clustering methods and divisive hierarchical clustering methods according to the creation of the dendrogram.

In agglomerative hierarchical clustering methods, each observation is initially treated as an independent cluster, and then repeatedly, until each individual observation obtains a single set of all observations, thereby forming a cluster with the closest observation.

In the divisive hierarchical clustering methods, initially all observations are evaluated as a single cluster and then repeatedly separated in such a way that each observation is separated from the farthest observation to form a new cluster. This process continues until all the observations create a single cluster.

Hierarchical clustering not only forms a group of similar series but also provides a graphical representation of the data. Graphical presentation allows the user to have an overall view of the data and an idea of data distribution. However, a small change in the data set leads to large changes in the hierarchical dendrogram. Another drawback is high computational complexity.

2.3.3 Density-based clustering

The density-based clustering approach is based on the concepts of density and attraction of objects. The idea is to create clusters of dense multi-dimensional areas where objects attract each other. In the core of dense areas, objects are very close together and crowded. The objects in the walls of the clusters were scattered less frequently than the core. In other words, density-based clustering determines dense areas of object space. The clusters are dense areas which are separated by rare dense areas. DBSCAN [51] and OPTICS [52] algorithms are the most known of density-based clustering examples.

The density-based approach is robust for noisy environments. The method also deals with outliers when defining embedded clusters. However, density-based clustering techniques cause difficulties due to high computational complexity and input parameter dependency when the dimensional index structure is not used.

2.3.4 Model-based clustering

The model-based approach [53–55] uses a statistical infrastructure to model the cluster structure of the time-series data. It is assumed that the underlying probability distributions of the data come from the final mixture. Model-based algorithms usually try to estimate the likelihood of the model parameters by applying some statistical techniques such as Expectation Maximization (EM). The EM algorithm iterates between an “E-step,” which computes a matrix z such that z_{ik} is an estimate of the conditional probability that observation i belongs to group k given the current parameter estimates, and an “M-step,” which computes maximum likelihood parameter estimates given z. Each data object is assigned to a cluster with the highest probability until the EM algorithm converges, so as to maximize likelihood for the entirety of the grant.

The most important advantage of the model-based approach is to estimate the probability that i. observation belongs to k. cluster. In some cases, the time series is likely to belong to more than one cluster. For such time-series data, the probability-giving function of the approach is the reason for preference. In this approach, it is assumed that the data set has a certain distribution but this assumption is not always correct.

2.3.5 Grid-based clustering

In this approach, grids made up of square cells are used to examine the data space. It is independent of the number of objects in the database due to the used grid structure. The most typical example is STING [56], which uses various levels of quadrilateral cells at different levels of resolution. It precalculates and records statistical information about the properties of each cell. The query process usually begins with a high-level hierarchical structure. For each cell at the current level, the confidence interval, which reflects the cell’s query relation, is computed. Unrelated cells are exempt from the next steps. The query process continues for the corresponding cells in the lower level until reaching the lowest layer.

After analyzing the data set and obtaining the clustering solution, there is no guarantee of the significance and reliability of the results. The data will be clustered even if there is no natural grouping. Therefore, whether the clustering solution obtained is different from the random solution should be determined by applying some tests. Some methods developed to test the quality of clustering solutions are classified into two types: external index and internal index.

- The external index is the most commonly used clustering evaluation method also known as external validation, external criterion. The ground truth is the goal clusters, usually created by experts. This index measures how well the target clusters and the resulting clusters overlap. Entropy, Adjusted Rand Index (ARI), F-measure, Jaccard Score, Fowlkes and Mallows Index (FM), and Cluster Similarity Measure (CSM) are the most known external indexes.

- The internal indexes evaluate clustering results using the features of data sets and meta-data without any external information. These are often used in cases where the correct solutions are not known. Sum of squared error is one of the most used internal methods which the distance to the nearest cluster determines the error. So clusters with similar time series are expected to give lower error values. Distance between two clusters (CD) index, root-mean-square standard deviation (RMSSTD), Silhouette index, R-squared index, Hubert-Levin index, semi-partial R-squared (SPR) index, weighted inter-intra index, homogeneity index, and separation index are the common internal indexes.

2.3.6 Clustering algorithm example: FunFEM

The funFEM algorithm [55, 57] allows to cluster time series or, more generally, functional data. FunFem is based on a discriminative functional mixture model (DFM) which allows the clustering of the curves (data) in a functional subspace. If the observed curves are $\{x_1, x_2 \ldots x_n\}$, FunFem aims cluster into K homogenous groups. It assumes that there exists an unobserved random variable Z = $\{z_1, z_2 \ldots z_n\}$ $\in \{0, 1\}^k$, if x belongs to group k, Z_k is defined as 1 otherwise 0. The clustering task goal is to predict the value $z_i = (z_{i1}, \ldots z_{ik})$ of Z for each observed curve x_i, for i = 1…n. The FunFem algorithm alternates, over the three steps of Fisher EM algorithm [57] ("F-step," "E-Step" and "M-step") to decide group memberships of Z = $\{z_1, z_2 \ldots z_n\}$. In other words, from 12 defined discriminative functional mixture (DFM) models, Fisher-EM decides which data fit the best. The Fisher-EM algorithm alternates between three steps:

- an E step in which posterior probabilities that observations belong to the K groups are computed,

- an F step that estimates the orientation matrix U of the discriminative latent space conditionally to the posterior probabilities,

- an M step in which parameters of the mixture model are estimated in the latent subspace by maximizing the conditional expectation of the complete likelihood.

Fisher-EM algorithm updates the parameters repeatedly until the Aitken criterion is provided. Aitken criterion estimates the asymptotic maximum of the log-likelihood in order to detect in advance the algorithm converge [57]. In model-based clustering, a model is defined by its number of component/cluster K and its parameterization. In model selection task, several models are reviewed while selecting the most appropriate model for the considered data.

FunFEM allows to choose between AIC (Akaike Information Criterion) [58], BIC (Bayesian information criteria) [59], and ICL (Integrated Completed Likelihood) [60] when deciding the number of clusters. The penalty terms are: $\frac{\gamma(M)}{2}$ $\log(n)$ in the BIC criterion, $\gamma(M)$ in the AIC criterion, and $\sum_{i=1}^{n}\sum_{k=1}^{K}t_{ik}\log(t_{ik})$ in the ICL criterion. Here, M indicates the number of parameters in the model, n is the number of observations, K is the number of clusters, and t_{ik} is the probability of ith observation belonging to kth cluster.

FunFem is implemented in R programming languages and serves as a function [61]. The algorithm is applied on a time series gene expression data in the following section. Input of the algorithm is gene expression data which is given in **Table 1**. The table shows the gene expression values measured as a result of the microarray experiment. The measurement was performed at six different times for each gene. The data were taken from the GEO database (GSE2241) [62]. FunFEM method is decided, and the best model is DkBk with $K = 4$ (bic = −152654.5) for input data. As a result, method assigned each gene to the appropriate cluster which is determined by the algorithm. **Table 2** demonstrates the gene symbol and cluster number. As a result, method assigned each gene to the appropriate cluster which is determined by the algorithm (**Table 2**).

Gene Symbol	TP1	TP2	TP3	TP4	TP5	TP6
AADAC	18.4	29.7	30	79.7	86.7	163.2
AAK1	253.2	141.8	49.2	118.7	145.2	126.7
AAMP	490	340.9	109.1	198.4	210.5	212
AANAT	5.6	1.4	3.7	3.1	1.6	4.9
AARS	1770	793.6	226.5	1008.9	713.3	1253.7
AASDHPPT	940.1	570.5	167.2	268.6	683	263.5
AASS	10.9	1.9	1.5	4.1	19.7	25.5
AATF	543.4	520.1	114.5	305.7	354.2	384.9
AATK	124.5	74.5	17	25.6	64.6	13.6
.	.	.	.	.	.	.
.	.	.	.	.	.	.
ZP2	4.1	1.4	0.8	1.4	1.4	3
ZPBP	23.4	13.7	7	7.8	22.3	26.9
ZW10	517.1	374.5	72.6	240.8	345.7	333.1
ZWINT	1245.4	983.4	495.3	597.4	1074.3	620.7
ZYX	721.6	554.9	135.5	631.5	330.9	706.8
ZZEF1	90.5	49.3	18.6	66.7	10.4	52.2
ZZZ3	457.3	317.1	93	243.2	657.5	443

Table 1.
Input data of the FunFEM algorithm.

Gene symbol	Cluster number
AADAC	2
AAK1	3
AAMP	3
AANAT	1
AARS	4
AASDHPPT	3
AASS	1
AATF	3
AATK	2
.	.
.	.
ZP2	1
ZPBP	1
ZW10	3
ZWINT	4
ZYX	4
ZZEF1	2
ZZZ3	3

Table 2.
Output data of the FunFEM algorithm.

3. Clustering approaches for gene expression data clustering

The approach to be taken depends on the application area and the characteristics of the data. For this reason, as a case study, the clustering of gene expression data, which is a special area of clustering of time-series data, will be examined in this section. Microarray is the technology which measures the expression levels of large numbers of genes simultaneously. DNA microarray technology overcomes traditional approaches in the identification of gene copies in a genome, in the identification of nucleotide polymorphisms and mutations, and in the discovery and development of new drugs. It is used as a diagnostic tool for diseases. DNA microarrays are widely used to classify gene expression changes in cancer cells.

The gene expression time series (gene profile) is a set of data generated by measuring expression levels at different cases/times in a single sample. Gene expression time series have two main characteristics, short and unevenly sampled. In The Stanford Microarray database, more than 80% of the time-series experiments contains less than 9 time points [63]. Observations below 50 are considered to be quite short for statistical analysis. Gene expression time-series data are separated from other time-series data by this characteristics (business, finance, etc.). In addition to these characteristics, three basic similarity requirements can be identified for the gene expression time series: scaling and shifting, unevenly distributed sampling points, and shape (internal structure) [64]. Scaling and shifting problems arise due to two reasons: (i) the expression of genes with a common sequence is similar, but in this case, the genes need not have the same level of expression at the same time. (ii) Microarray technology, which is often corrected by normalization.

The scaling and shifting factor in the expression level may hide similar expressions and should not be taken into account when measuring the similarity between the two expression profiles. Sampling interval length is informative and cannot be ignored in similarity comparisons. In microarray experiments, the density change characterizes the shape of the expression profile rather than the density of the gene expression. The internal structure can be represented by deterministic function, symbols describing the series, or statistical models.

There are many popular clustering techniques for gene expression data. The common goal of all is to explain the different functional roles of the genes that play a key biological process. Genes expressed in a similar way may have a similar functional role in the process [65].

In addition to all these approaches, it is possible to examine the cluster of gene expression data in three different classes as gene-based clustering, sample-based clustering, and subspace clustering (**Figure 2**) [66]. In gene-based clustering, genes are treated as objects, instances (time-point/patient-intact) as features. Sample-based clustering is exactly the opposite: samples are treated as objects, genes as features. The distinction between these two clustering approaches is based on the basic characterization of the clustering process used for gene expression data. Some clustering algorithms, such as K-means and hierarchical approach, can be used to cluster both genes and fragments of samples. In the molecular biology, "any function in the cell is carried out with the participation of a small subset of genes, and the cellular function only occurs on a small sample subset." With this idea, genes and samples are handled symmetrically in subspace clustering; gene or sample, object or features.

In **gene-based clustering**, the aim is to group the co-expressed genes together. However, due to the complex nature of microarray experiments, gene expression data often contain high amounts of noise, characterizing features such as gene expression data often linked to each other (clusters often have a high intersection ratio), and some problems arising from constraints from the biological domain.

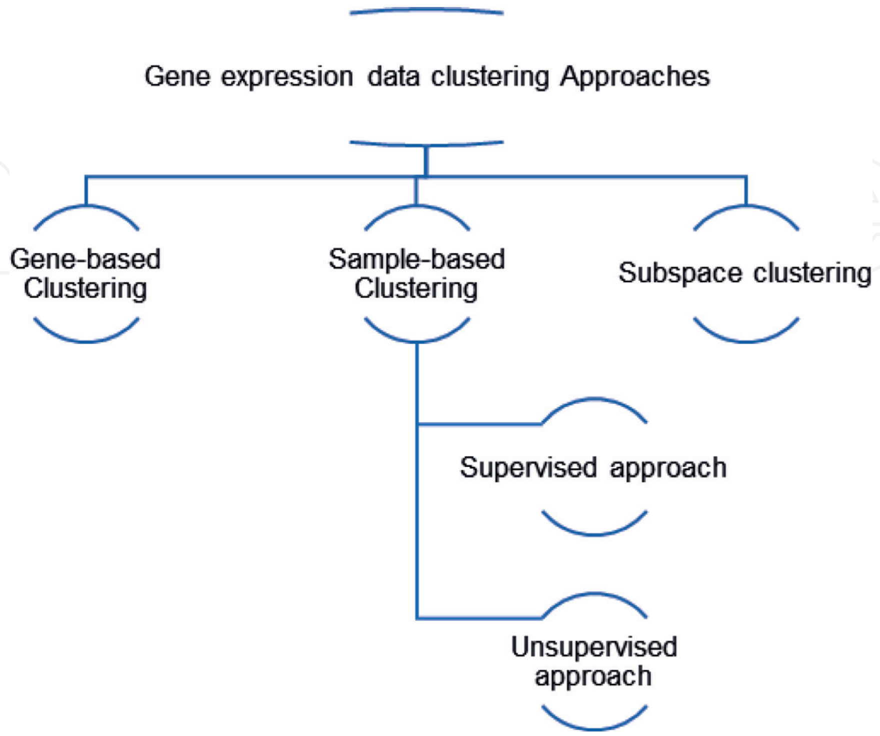

Figure 2.
Gene expression data clustering approaches.

Also, among biologists who will use microarray data, the relationship between genes or clusters that are usually related to each other within the cluster, rather than the clusters of genes, is a more favorite subject. That is, it is also important for the algorithm to make graphical presentations not just clusters. K-means, self-organizing maps (SOM), hierarchical clustering, graph-theoretic approach, model-based clustering, and density-based approach (DHC) are the examples of gene-based clustering algorithms.

The goal of the **sample-based approach** is to find the phenotype structure or the sub-structure of the sample. The phenotypes of the samples studied [67] can only be distinguished by small gene subsets whose expression levels are highly correlated with cluster discrimination. These genes are called informative genes. Other genes in the expression matrix have no role in the decomposition of the samples and are considered noise in the database. Traditional clustering algorithms, such as K-means, SOM, and hierarchical clustering, can be applied directly to clustering samples taking all genes as features. The ratio of the promoter genes to the nonrelated genes (noise ratio) is usually 1:10. This also hinders the reliability of the clustering algorithm. These methods are used to identify the informative genes. Selection of the informative genes is examined in two different categories as supervised and unsupervised. The supervised approach is used in cases where phenotype information such as "patient" and "healthy" is added. In this example, the classifier containing only the informative genes is constructed using this information. The supervised approach is often used by biologists to identify informative genes. In the unsupervised approach, no label specifying the phenotype of the samples is placed. The lack of labeling and therefore the fact that the informative genes do not guide clustering makes the unsupervised approach more complicated. There are two problems that need to be addressed in the unsupervised approach: (i) the high number of genes versus the limited number of samples and (ii) the vast majority of collected genes are irrelevant. Two strategies can be mentioned for these problems in the unsupervised approach: unsupervised gene selection and clustering. In unsupervised gene selection, gene selection and sample clustering are treated as two separate processes. First, the gene size is reduced, and then classical clustering algorithms are applied. Since there is no training set, the choice of gene is based solely on statistical models that analyze the variance of gene expression data. Associated clustering dynamically supports the combination of repetitive clustering and gene selection processes by the use of the relationship between genes and samples. After many repetitions, the sample fragments converge to the real sample structure and the selected genes are likely candidates for the informative gene cluster.

When **subspace clustering** is applied to gene expression vectors, it is treated as a "block" consisting of clusters of genes and subclasses of experimental conditions. The expression pattern of the genes in the same block is consistent under the condition in that block. Different greedy heuristic approaches have been adapted to approximate optimal solution.

Subspace clustering was first described by Agrawal et al. in 1998 on general data mining [68]. In subspace clustering, two subspace sets may share the same objects and properties, while some objects may not belong to any subspace set. Subspace clustering methods usually define a model to determine the target block and then search in the gen-sample space. Some examples of subspatial cluster methods proposed for gene expression are biclustering [69], coupled two way clustering (CTWC) [70], and plaid model [71].

According to different clustering criteria, data can be clustered such as the co-expressing gene groups, the samples belonging to the same phenotype or genes from the same biological process. However, even if the same criteria are used in different clustering algorithms, the data can be clustered in different forms. For this reason, it is necessary to select more suitable algorithm for data distribution.

4. Conclusions

Clustering for time-series data is used as an effective method for data analysis of many areas from social media usage and financial data to bioinformatics. There are various methods introduced for time-series data. Which approach is chosen is specific to the application. The application is determined by the needs such as time, speed, reliability, storage, and so on. When determining the approach to clustering, three basic issues need to be decided: data representation, similarity measure, and clustering algorithm.

The data representation involves transforming the multi-dimensional and noisy structure of the time-series data into a less dimensional that best expresses the whole data. The most commonly used method for this purpose is dimension reduction or feature extraction.

It is challenging to measure the similarity of two time series. The chapter has been examined similarity measures in three sections as similarity in shape, similarity in time, and similarity in change.

For the time-series clustering algorithms, it is not wrong to say that the evolution of conventional clustering algorithms. Therefore, the classification of traditional clustering algorithms (developed for static data) has been included. It is classified as partitioning, hierarchical, model-based, grid-based, and density-based. Partition algorithms initially require prototypes. The accuracy of the algorithm depends on the defined prototype and updated method. However, they are successful in finding similar series and clustering time series with equal length. The fact that the number of clusters is not given as the initial parameter is a prominent and well-known feature of hierarchical algorithms. At the same time, works on time series that are not of equal length causes it to be one step ahead of other algorithms. However, hierarchical algorithms are not suitable for large data sets due to the complexity of the calculation and the scalability problem. Model-based algorithms suffer from problems such as initialization of parameters based on user predictions and slow processing time for large databases. Density-based algorithms are not generally preferred over time-series data due to their high working complexity. Each approach has pros and cons compared to each other, and the choice of algorithm for time-series clustering varies completely according to the characteristics of the data and the needs of the application. Therefore, in the last chapter, a study on the clustering of gene expression data, which is a specific field of application, has been mentioned.

In time-series data clustering, there is a need for algorithms that execute fast, accurate, and with less memory on large data sets that can meet today's needs.

Author details

Esma Ergüner Özkoç
Başkent University, Ankara, Turkey

*Address all correspondence to: eeozkoc@baskent.edu.tr

References

[1] Ratanamahatana C. Multimedia retrieval using time series representation and relevance feedback. In: Proceedings of 8th International Conference on Asian Digital Libraries (ICADL2005); 2005. pp. 400-405

[2] Özkoç EE, Oğul H. Content-based search on time-series microarray databases using clustering-based fingerprints. Current Bioinformatics. 2017;**12**(5):398-405. ISSN: 2212-392X

[3] Lin J, Keogh E, Lonardi S, Lankford J, Nystrom D. Visually mining and monitoring massive time series. In: Proceedings of 2004 ACM SIGKDD International Conference on Knowledge Discovery and data Mining–KDD '04; 2004. p. 460

[4] Bornemann L, Bleifuß T, Kalashnikov D, Naumann F, Srivastava D. Data change exploration using time series clustering. Datenbank-Spektrum. 2018;**18**(2):79-87

[5] Rani S, Sikka G. Recent techniques of clustering of time series data: A survey. International Journal of Computers and Applications. 2012;**52**(15):1

[6] Aghabozorgi S, Shirkhorshidi AS, Wah TY. Time-series clustering–A decade review. Information Systems. 2015;**53**:16-38

[7] Lin J, Keogh E, Lonardi S, Chiu B. 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; 13 June 2003; ACM; pp. 2-11

[8] Keogh EJ, Pazzani MJ. A simple dimensionality reduction technique for fast similarity search in large time series databases. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining; 18 April 2000; Springer, Berlin, Heidelberg. pp. 122-133

[9] Esling P, Agon C. Time-series data mining. ACM Computing Surveys (CSUR). 2012;**45**(1):12

[10] Keogh E, Lin J, Fu A. Hot sax: Efficiently finding the most unusual time series subsequence. In: Fifth IEEE International Conference on Data Mining (ICDM'05); 27 November 2005; IEEE. pp. 226-233

[11] Ghysels E, Santa-Clara P, Valkanov R. Predicting volatility: Getting the most out of return data sampled at different frequencies. Journal of Econometrics. 2006;**131**(1-2):59-95

[12] Kawagoe GD. Grid Representation of Time Series Data for Similarity Search. In: Data Engineering Workshop; 2006

[13] Agronomischer Zeitreihen CA. Time Series Clustering in the Field of Agronomy. Technische Universitat Darmstadt (Master-Thesis); 2013

[14] Keogh E, Lonardi S, Ratanamahatana C. Towards parameter-free data mining. In: Proceedings of Tenth ACM SIGKDD International Conference on Knowledge Discovery Data Mining; 2004, Vol. 22, No. 25. pp. 206-215

[15] Keogh E, Chakrabarti K, Pazzani M, Mehrotra S. Locally adaptive dimensionality reduction for indexing large time series databases. ACM SIGMOD Record. 2001;**27**(2):151-162

[16] Keogh E, Pazzani M. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In: Proceedings of the 4th International

Conference of Knowledge Discovery and Data Mining; 1998. pp. 239-241

[17] Korn F, Jagadish HV, Faloutsos C. Efficientlysupportingadhoc queries in large datasets of time sequences. ACM SIGMOD Record. 1997;**26**: 289-300

[18] Faloutsos C, Ranganathan M, Manolopoulos Y. Fasts ubsequence matching in time-series databases. ACM SIGMOD Record. 1994;**23**(2): 419-429

[19] Portet F, Reiter E, Gatt A, Hunter J, Sripada S, Freer Y, et al. Automatic generation of textual summaries from neonatal intensive care data. Artificial Intelligence. 2009;**173**(7):789-816

[20] Chan K, Fu AW. Efficient time series matching by wavelets. In: Proceedings of 1999 15th International Conference on Data Engineering; 1999, Vol. 15, no. 3. pp. 126-133

[21] Agrawal R, Faloutsos C, Swami A. Efficient similarity search in sequence databases. Foundations of Data Organization and Algorithms. 1993;**46**: 69-84

[22] Kawagoe K, Ueda T. A similarity search method of time series data with combination of Fourier and wavelet transforms. In: Proceedings Ninth International Symposium on Temporal Representation and Reasoning; 2002. pp. 86-92

[23] Chung FL, Fu TC, Luk R. Flexible time series pattern matching based on perceptually important points. In: Jt. Conference on Artificial Intelligence Workshop. 2001. pp. 1-7

[24] Keogh E, Pazzani M, Chakrabarti K, Mehrotra S. A simple dimensionality reduction technique for fast similarity search in large time series databases. Knowledge and Information Systems. 2000;**1805**(1):122-133

[25] Caiand Y, Ng R. Indexing spatio-temporal trajectories with Chebyshev polynomials. In: Procedings of 2004 ACM SIGMOD International; 2004. p. 599

[26] Bingham E. Random projection in dimensionality reduction: Applications to image and text data. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; 2001. pp. 245-250

[27] Chen Q, Chen L, Lian X, Liu Y. Indexable PLA for efficient similarity search. In: Proceedings of the 33rd International Conference on Very large Data Bases; 2007. pp. 435-446

[28] Corduas M, Piccolo D. Timeseries clustering and classification by the autoregressive metric. Computational Statistics & Data Analysis. 2008;**52**(4): 1860-1872

[29] Kalpakis K, Gada D, Puttagunta V. Distance measures for effective clustering of ARIMA time-series. In: Proceedings 2001 IEEE International Conference on Data Mining; 2001. pp. 273-280

[30] Kumar N, Lolla N, Keogh E, Lonardi S. Time-series bitmaps: A practical visualization tool for working with large time series databases. In: Proceedings of the 2005 SIAM International Conference on Data Mining; 2005. pp. 531-535

[31] Minnen D, Starner T, Essa M, Isbell C. Discovering characteristic actions from on body sensor data. In: Proceedings of 10th IEEE International Symposium on Wearable Computers; 2006. pp. 11-18

[32] Minnen D, Isbell CL, Essa I, Starner T. Discovering multivariate motifs using

subsequence density estimation and greedy mixture learning. In: Proceedings of the National Conference on Artificial Intelligence; 2007, Vol. 22, No. 1. p. 615

[33] Panuccio A, Bicego M, Murino V. A hidden Markov model-based approach to sequential data clustering. In: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR). Berlin, Heidelberg: Springer; 2002, pp. 734-743

[34] Bagnall AAJ, "Ann" Ratanamahatana C, Keogh E, Lonardi S, Janacek G. A bit level representation for time series data mining with shape based similarity. Data Mining and Knowledge Discovery. 2006;**13**(1): 11-40

[35] Ratanamahatana C, Keogh E, Bagnall AJ, Lonardi S. A novel bit level time series representation with implications for similarity search and clustering. In: Proceedings of 9th Pacific-Asian International Conference on Knowledge Discovery and Data Mining (PAKDD'05); 2005. pp. 771-777

[36] Bagnall AJ, Janacek G. Clustering time series with clipped data. Machine Learning. 2005;**58**(2):151-178

[37] Sakoe H, Chiba S. A dynamic programming approach to continuous speech recognition. In: Proceedings of the Seventh International Congress on Acousticsvol; 1971, Vol. 3. pp. 65-69

[38] Sakoe H, Chiba S. Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1978;**26**(1):43-49

[39] Smyth P. Clustering sequences with hidden Markov models. Advances in Neural Information Processing Systems. 1997;**9**:648-654

[40] Xiong Y, Yeung DY. Mixtures of ARMA models for model-based time series clustering. In: Data Mining, 2002. ICDM 2003; 2002. pp. 717-720

[41] Han J, Kamber M. Data Mining: Concepts and Techniques. San Francisco: Morgan Kaufmann; 2001. pp. 346-389

[42] Liao TW. Clustering of time series data—a survey. Pattern Recognition. 2005;**38**(11):1857-1874

[43] MacQueen J. Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability; 21 June 1967, Vol. 1, No. 14. pp. 281-297

[44] Tamayo P, Slonim D, Mesirov J, Zhu Q, Kitareewan S, Dmitrovsky E, et al. Interpreting patterns of gene expression with self-organizing maps: Methods and application to hematopoietic differentiation. Proceedings of the National Academy of Sciences. 1999;**96**(6):2907-2912

[45] Ng RT, Han J. Efficient and effective clustering methods for spatial data mining. In: Proceedings of the International Conference on Very Large Data Bases; 1994. pp. 144-144

[46] Kaufman L, Rousseeuw PJ, Corporation E. Finding Groups in Data: An Introduction to Cluster Analysis, Vol. 39. Hoboken, NewJersey: Wiley Online Library; 1990

[47] Guha S, Rastogi R, Shim K. CURE: An efficient clustering algorithm for large databases. ACM SIGMOD Record. 1998;**27**(2):73-84

[48] Zhang T, Ramakrishnan R, Livny M. BIRCH: An efficient data clustering method for very large databases. ACM SIGMOD Record. 1996;**25**(2): 103-114

[49] Karypis G, Han EH, Kumar V. Chameleon: Hierarchical clustering using dynamic modeling. Computer. 1999;**32**(8):68-75

[50] Beal M, Krishnamurthy P. Gene expression time course clustering with countably infinite hidden Markov models. arXiv preprint arXiv:1206.6824; 2012

[51] Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial data bases with noise. In: Knowledge Discovery and Data Mining. Vol. 96, No. 34; August 1996. pp. 226-231

[52] Ankerst M, Breunig M, Kriegel H. OPTICS: Ordering points to identify the clustering structure. ACM SIGMOD Record. 1999;**28**(2):40-60

[53] Fisher DH. Knowledge acquisition via incremental conceptual clustering. Machine Learning. 1987;**2**(2):139-172

[54] Carpenter GA, Grossberg S. A massively parallel architecture for a self-organizing neural pattern recognition machine. Computer Vision Graphics Image Process. 1987;**37**(1):54-115

[55] Bouveyron C, Côme E, Jacques J. The discriminative functional mixture model for the analysis of bike sharing systems. The Annals of Applied Statistics. 2015;**9**(4):1726-1760

[56] Wang W, Yang J, Muntz R. STING: A statistical information grid approach to spatial data mining. In: Proceedings of the International Conference on Very Large Data Bases; 1997. pp. 186-195

[57] Bouveyron C, Brunet C. Simultaneous model-based clustering and visualization in the fisher discriminative subspace. Statistics and Computing. 2012;**22**:301-324

[58] Akaike H. A new look at the statistical model identification. IEEE Transactions on Automatic Control. 1974;**19**:716-723

[59] Kass RE, Raftery AE. Bayes factors. Journal of the American Statistical Association. 1995;**90**(430):773-795

[60] Biernacki C, Celeux G, Govaert G. Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000;**22**:719-725

[61] Bouveyron C. funFEM: Clustering in the Discriminative Functional Subspace. R package version. 2015;1

[62] Barrett T, Troup DB, Wilhite SE, Ledoux P, Rudnev D, Evangelista C, et al. NCBI GEO: Archive for high-throughput functional genomic data. Nucleic Acids Research. 2009;**37** (Database):D885-D890

[63] Kuenzel L. Gene clustering methods for time series microarray data. Biochemistry. 2010;**218**

[64] Moller-Levet CS, Cho KH, Yin H, Wolkenhauer O. Clustering of gene expression time-series data. Technical report. Department of Computer Science, University of Rostock, Germany; 2003

[65] Beal M, Krishneamurthy P. Gene expression time course clustering with countably infinite hidden Markov models. arXiv preprint arXiv:1206.6824; 2012

[66] Jiang D, Tang C, Zhang A. Cluster analysis for gene expression data: A survey. IEEE Transactions on Knowledge and Data Engineering. 2004; **16**(11):1370-1386

[67] Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, Mesirov JP, et al. Molecular classification of cancer: Class discovery and class prediction by

gene expression monitoring. Science. 1999;**286**(5439):531-537

[68] Agrawal R, Gehrke J, Gunopulos D, Raghavan P. Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications. ACM; 1998; **27**(2):94-105

[69] Cheng Y, Church GM. Biclustering of expression data. In: ISMB; 2000, Vol. 8, No. 2000. pp. 93-103

[70] Getz G, Levine E, Domany E. Coupled two-way clustering analysis of gene microarray data. Proceedings of the National Academy of Sciences. 2000;**97**(22):12079-12084

[71] Lazzeroni L, Owen A. Plaid models for gene expression data. Statistica Sinica. 2002;**1**:61-86