Researches Advanced in Clustering Algorithms

Guohuan Feng[1, †], Junchen Lin[2, †] and Keyi Wang[3, *, †]

[1]College of software, Zhengzhou University of Light Industry, Zhengzhou, China

[2]College of Computer Science and Network Engineering, Guangzhou University, Guangzhou, China

[3]School of Aeronautics and Astronautics, Zhejiang University, Zhejiang, China

*Corresponding author: 3180103358@zju.edu.cn

†These authors contributed equally.

Abstract. Clustering is a technique to find the intrinsic structure between data and is a fundamental problem in many data-driven application fields. Currently, clustering is generally modeled as an unsupervised learning task, aiming to mine similar features between different samples and cluster samples with similar features into clusters. Ideally, objects in the same cluster are expected to be similar in the clustering results, while objects in different clusters are quite different. This study summarizes the research status of clustering algorithms in recent years. Specifically, the relevant critical steps of clustering algorithms are first introduced. From two aspects of partition and hierarchical clustering, representative clustering algorithms such as K-means, K-medoids, CLARANS, BIRCH, DBSCAN, and CURE are further detailed. This study also analyzes and summarizes the above algorithms in terms of critical technologies, algorithm ideas, benefits, and shortcomings and compares the distance accuracy of different algorithms on standard data sets. The above work will provide a valuable reference for cluster analysis and data mining research.

Keywords: Clustering; unsupervised learning; feature representation.

1. Introduction

In different environments, people often need to group things to facilitate management. People label things with the same or similar characteristics and put them into groups, while things without similar characteristics are classified into different groups. One needs to ensure that the grouping is correct so that nothing is placed in the wrong group. Grouping an already-tagged thing is easy because the tag is given in advance. However, in many cases, the labels are not given in advance, and we need to extract the similarities of various things to group them. This grouping of unlabeled things is called unsupervised classification, and the process of extracting and learning the features of these things is called unsupervised learning. It is one of the most commonly used unsupervised learning algorithms. The clustering challenge is a fundamental issue in machine learning, computer vision, data compression, and other research. Also, it plays an incredibly significant role in identifying data's inherent structure.

The study of cluster analysis has a lengthy history, and its significance and intersecting properties with other research areas were recognized decades ago. Clustering algorithms in pattern recognition are mainly used in speech and character recognition. In machine learning, clustering algorithms are used in image segmentation and machine vision, mainly for data compression and information retrieval in image processing. In addition, the essential clustering applications are data mining, also called multi-relational data mining, and spatiotemporal database applications, such as GIS. Moreover, sequence and heterogeneous data analysis are also essential applications of clustering. Conventional clustering methods have been relatively well developed. However, the number and complexity of stored data have increased due to the geometric growth of human activities, and the difficulty of clustering tasks has increased as the connections between data and the characteristics of the data itself have become more complicated. With the improvement, it has faced problems such as high computational complexity, low dimensionality degradation effect, and decreased clustering effect. The importance of data lies in the data itself and in a series of analytical activities based on

the data, helping people to discover underlying information and produce meaningful, in-depth information.

After a long development, unsupervised learning has relatively perfect research ideas and component analysis methods, and the accuracy is also constantly improving. This paper focuses on clustering algorithms in the field of unsupervised learning algorithms. The first half introduces the basic concept of clustering algorithm and feature extraction. The central part discusses two clustering algorithms based on different classification methods: the partition clustering algorithm and the hierarchical clustering algorithm. In the partition clustering algorithm, we focus on three classical algorithms: K-means, K-medoids, and CLARANS, and in the hierarchical clustering algorithm, we focus on BIRCH, DBSCAN, and CURE, and their advantages and limitations are compared.

In this paper, three classical algorithms are selected in the partition and hierarchical clustering algorithms, and their applicability, limitations, efficiency, parameters, and so on are discussed. Finally, a summary table shows the differences between various algorithms. At the end of the paper, a summary and some views on the future development trend are summarized.

2. Representation Clustering Teachniques

2.1. Key Steps of Clutering

2.1.1 Feature Representation

As a way of data classification, the main difference between clustering and classification is that clustering is an unsupervised learning algorithm, while classification belongs to a supervised learning algorithm. As a typical unsupervised learning algorithm, clustering is mainly used to automatically classify similar samples into a category.

Representation is one of the core themes of deep learning. A classic unsupervised learning task is to find the best representation of data, remove irrelevant factors that do not affect the overall situation or have minimal influence factors, and find the most core and simple critical representation of data. Here, the simple representation includes low latitude, sparse, and independent representation.

(1)Low latitude Representation. Compress the information in X into a more miniature representation as much as possible, which usually produces elements with smaller or weaker dependencies than the original high-dimensional data.

(2)Sparse Representation. Embedding the data set into the representation with most input items being zero is usually used when the dimension needs to be increased so that most of the zero representations will not lose a lot of information.

(3)Independent Representation. Try to separate the sources of changes in data distribution so that the representation dimensions are independent.

2.1.2 Feature Dimension Reduction

Feature dimensionality reduction refers to reducing the number of random variables to obtain a group of "uncorrelated" principal variables under certain conditions. Two commonly used dimensionality reduction methods include feature selection and principal component analysis. Feature selection aims to find the main features from the original ones because we are all valuable features for learning during training. If there are problems in the features themselves or the correlation between features is strong, it will significantly impact algorithm learning and prediction.

The methods of feature selection include filtering and embedding. Filter mainly explores the characteristics of the feature itself, the relationship between the feature and the feature and the target value. Standard used filter methods include the variance selection method such as low variance feature filtering, correlation coefficient, etc. For the embedding methods, which aim to automatically select features according to the correlation between features and target values, the standard methods mainly include the decision tree, information entropy, information gain, regularization technologies and convolutional neural network.The methods of feature selection include filtering and embedding.

2.1.3 Similarity Measurement

After clustering, the similarity between objects in a cluster is an important criterion to judge whether a cluster is good or bad. In the same cluster, the greater the similarity between objects, the better the clustering effect. The similarity between objects is mainly determined by the distance between them. Many clustering methods use distance measurement to determine the similarity and dissimilarity between any two objects.

2.2. Partitioned Clustering

According to the different clustering methods, the two standard clustering algorithms are divided clustering algorithm and hierarchical clustering algorithm. The partition clustering algorithm divides the data set into k parts by optimizing the evaluation function, which requires K as the input parameter[2]. The most representative partitioned clustering algorithms[3] includes the K-NN[4], K-medoids and CLARANS[5], which will be introduced in detail step by step.

2.2.1 K-NN

The k-nearest neighbor algorithm (K-NN) is a supervised learning algorithm, which is nonparametric. Its input is composed of the k closest training instances in the dataset and the output is contributed to whether the K-NN is intended for classification or regression.[6] If K-NN is used for classification, the output will be a class. The object's class will be defined by the vote of its neighbors, to be precise, the algorithm will count the number of each neighbor's vote, and the object is allocated to a class, which is most commonly appear among its K nearest neighbors , where K is a positive and small integer. If K = 1, the only nearest neighbor's class will be allocated to the object. If K-NN is used for regression, the output will be the attribute value of the object, which is the average of the K nearest neighbors' values.

K-NN is an algorithm that relies on distance for classification, and therefore its accuracy can be significantly improved if distinct features stand for different types of physical units or have sunstantial variations[7,8]. Euclidean distance is frequently used to represent distance if variables are continuous. However, if variables are discrete, such as classifying text or classifying pictures, an alternative metric, the overlap metric (or Hamming distance) can be applied. For example, in the circumstance of gene expression microarray data, K-NN utilizes correlation coefficients as metrics. Usually, being learned with specific algorithms such as great marginal nearest neighbor or neighborhood component analysis, can benefit the distance metrics and thus lead to significant improvement of classification accuracy of K-NN. For classification and regression, weights can be assigned to the neighbors' contributions so that near neighbors contribute more to the mean than far neighbors. As d is defined as the distance away from the neighbor, the typical solution to weight each neighbor is to assign them a weight of 1/d.

The neighbors come from a group of objects with known category (for K-NN classification) or object attribute values (for K-NN regression). This can be considered as the training set of the algorithm, where the training samples are vectors that each of them has already assign to a label of class and they are in a multidimensional feature space. Moreover, there are only feature vectors and training samples' labels in the training phase of the algorithm. In the case of classification, k is a constant that is defined by users and a vector without a label (a query point or test point) is classified by assigning the most frequent label of the k training samples to that query point.

2.2.2 K-medoids

The k-medoids algorithm is a clustering algorithm that resembles k-means[9]. K-means[9] and k-medoids algorithms are both local (dividing the data set into groups) and seek to minimize the distance between the dots marked as clusters and the points specified as the clusters' centres. Compared with the k-means algorithm, k-medoids select accurate data points to be centred and therefore have more interpretable cluster centres than k-means. Usually, the medoid of a cluster is considered as the one in the cluster with the least averaged divergence from all objects in the cluster, i.e., it is the cluster point that is closest to the centre, however as the cluster centre is determined by the average of the

cluster's points, it might not be one of the input data points. In addition, k-medoids can be used for discretionary similarity measures, while k-means usually demand Euclidean distance to obtain a reasonable solution. Since k-medoids minimize the sum of pairwise similarities rather than the sum of squares of Euclidean distances, it is more stable to noise and outliers than k-means. K-medoids is a conventional cluster partitioning technique that divides a dataset of n objects into k clusters, where the number of clusters k is assumed to be known in advance. In contrast, the particular value of k that is good or bad can be evaluated using methods such as the silhouette method to evaluate.

The specific implementation steps of the algorithm are as below. (1) K data samples are randomly selected as reference points O_j among the data samples (j = 1,2,3…. K). (2) Assign the remaining sample points to K cluster classes. (3) Randomly select a non-reference point sample O and swap O and O_1 reference points. Repeat the action in (2) to generate a new set of cluster classes. Calculate the objective function S. If S < 0, swap O and O_1 and keep the new cluster classes; otherwise, keep the original centroids and clusters. Repeat this step until K centroids no longer change.

2.2.3 CLARANS

The CLARANS algorithm combines the sampling technique (CLARA) with PAM. The main idea is that instead of considering the whole data set, a small part of the actual data is selected as a representative of the data. Then the PAM method is used to choose the centroids from the sample. If the sample is selected in a very unexpected way, then it should be close to representing the original data set. The representative object (centroid) selected from it is likely to be similar to the representative object selected from the entire data set.CLARA takes multiple samples of the data set, applies the PAM algorithm to each sample, and returns the best clustering result as the output.
The specific steps of the algorithm are as below.

(1) Input parameters numlocal and maxneighbor. numlocal indicates the number of samples, maxneighbor indicates the number of nodes that a node can compare with any particular neighbour Let: i=1, where i is used to indicating the number of samples that have been selected min-cost is the minimum cost, initially set to a great number.

(2) Make the current node current to any of the nodes in G_n.

(3) Set the value of j equal to 1, where j is used to indicate the number of neighbours that have been compared with the current.

(4) Consider a random neighbour S of the current point and calculate the cost difference between the two nodes. The surrogate value is defined by the sum of the dissimilarity (distance or similarity) between each object and its cluster centroid. The cost difference is the difference in cost between two random domains.

(5) If the cost of S is lower, the current node equals S and goes to step (3).

(6) Otherwise, let j=j+1. If j<= max neighbour, then goes to step (4).

(7) Otherwise, when j>maxneighbor, the current node is the minimum cost node for this selection. If its cost is less than the min-cost, let min-cost be the cost of the current node and the best node be the current node.

(8) Let i= i+1. if i > numlocal, output the best node and abort the whole operation. Otherwise, go to step (2).

2.3. Hierarchical Clustering

Hierarchical clustering[10] is composed of various levels of segmentation clustering with nested relationships between the levels. Compared to segmented clustering algorithms, its main advantage is that it does not require any input parameters, and its shortcoming is that termination conditions must be designated.

2.3.1 BIRCH

BIRCH is a comprehensive hierarchical clustering method whose basic idea is to compress data objects into many sub-clusters and then perform the clustering process on these sub-clusters[11]. The information about a cluster is represented using triples of -cluster features, allowing the representation

of a cluster point to be made up of the appropriate cluster characteristics rather than specific -cluster points. Clustering is obtained by constructing clustering feature trees that satisfy the constraints of branching factors and group diameters. A highly balanced tree with two parameter branch factors is the clustering feature tree (and class diameter). While the class diameter represents the restriction on the diameter size of a class of points, that is, the range within which these points may be collected into a class, the branching factor determines the maximum number of children per node of the tree. BIRCH could perform good clustering through one scan, which showed that the algorithm was suitable for large data volumes. BIRCH algorithm is only suitable for convex and spherical distribution of classes, and the BIRCH algorithm is not feasible for invisible high-dimensional data because the BIRCH algorithm needs to provide correct clustering number and cluster diameter restrictions. BIRCH algorithm flow is shown in figure 1 below:

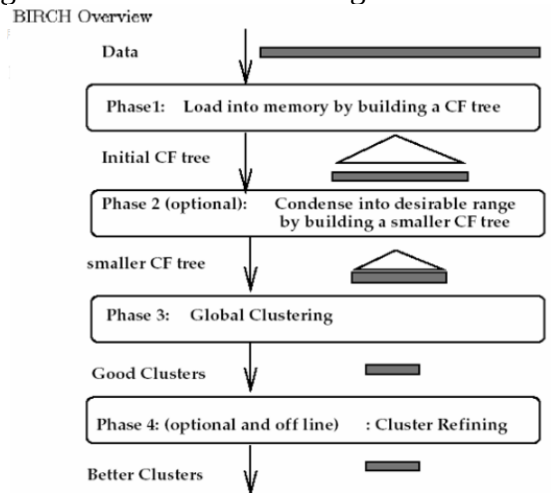


Figure 1 The overview of BIRCH algorithm

The whole methodology is implemented in 4 steps:

(1) Scan all the data, create the first CF tree, group the dense data into clusters, and consider the sparse data as a collection of isolated points.

(2) This phase is optional. Phase 3's global or semi-global clustering method needs input ranges to work quickly and accurately. In light of the previous step, this one creates a more compact CF tree.

(3) A global/semi-global technique is employed to cluster all leaf nodes in order to account for the split brought on by the input order and page size.

(4) This phase is likewise a choice. The data points are redistributed to the closest seeds using the stage 3 centroids as seeds to make sure that duplicates are placed in the same clusters and cluster names are added.

2.3.2 DBSCAN

Martin Ester develops DBSCAN(Density-based spatial clustering of applications with noise). Hans-peter Kriegel et al. proposed a density-based spatial data clustering method in 1996, the most commonly used method[11]. The algorithm takes a region with sufficient density as the distance centre. It keeps growing the region since a cluster can be uniquely determined by any central object in it [4]. The notion of density-based clustering is utilized by the algorithm, that is, the quantity of objects (points or other spatial objects) in a specific region of the clustering space does not fall below a given threshold. This method allows finding arbitrary clusters in a noisy spatial database and reduce the density.

A significant enough adjacent area connection can effectively deal with abnormal data, mainly used for spatial data clustering.

Pseudo-code of algorithm implementation:

Input sample data D(x1,x2... xn).

Parameter neighbourhood radius R and sample number threshold M were set.

Initialize the core point collection $\omega = \varnothing$.

Iterating each sample data to determine whether it is a core point is based on a neighbourhood drawing with r as the radius to determine whether the number of samples in the neighbourhood is more excellent than M. If it is, adds it to the core point set; otherwise, skip it.

The number of initialization clusters is k=0.

Iterate over the set of core points.

Then, all samples in the neighbourhood of the core point are added to the queue. If a neighbourhood is also a core point, all samples in its neighbourhood are also added to the queue until the queue is empty.

All of the samples that I just walked through are grouped into one cluster

And then iterate over the remaining core points in the OMEGA until all the core points have been traversed.

It can be said that each iteration of a large loop generates a cluster class, and steps 6-9 are very similar to BFS's (breadth-first search) implementation of queuing, enqueuing adjacent nodes, and then enqueuing the current node, continuously adding adjacent elements of each node in the paired column to the queue until the queue is empty and the iteration ends.

2.3.3 CURE

CURE(Chustrring Using Reprehensives) better solved the problem of preference for sphericity and similar size by Using a brand-new hierarchical clustering technique that chose a middle ground between the representative object-based approach and the basic centroid approach. It chooses a predetermined number of representative points from the data space rather than a single centroid or objects representing a cluster. Combining random sampling with partitioning, the technique divides the sample into smaller portions, with each partition being somewhat clustered, and these results are then grouped into groups. The CURE algorithm yields some interesting results: CURE can find exciting shapes, filter boundary points, sampling and partitioning can reduce input size, and for large, high-dimensional databases, the CURE time complexity is quadratic concerning sample size, independent of the overall database size. Although the CURE algorithm has the above advantages, the clustering results are sensitive to the input parameters, and expert knowledge is needed to set the appropriate parameter values.

CURE algorithm, the basic process can be expressed as the following figure (figure 2); the first part of randomly selected sample points from the original data set as a subset of the subset is divided again after the partition on the set of operation CURE clustering algorithm for each set of clusters, and remove the outliers, and then to CURE these cluster further hierarchical clustering, And delete the outliers, and finally divide the remaining data set sample points in the disk.

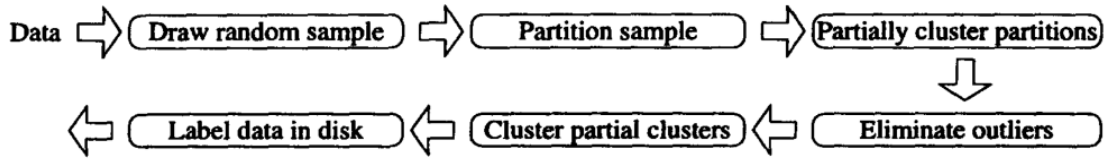


Figure 2 The basic process of CURE

The advantages of K-NN and K-medoids are their simplicity and ease of implementation. The disadvantages are their sensitivity to outliers, and sometimes, the centroids are not meaningful. The advantages of CLARANS are that it can handle outliers efficiently, and the disadvantages are its high cost.

BIRCH algorithm benefits fast clustering speed, noise point identification, and preliminary classification of data sets. However, the BIRCH algorithm has the disadvantages of poor performance for high-dimensional feature data and non-convex data sets and may lead to different classification distribution from accurate data. The advantage of the DBSCAN clustering algorithm is that it does not need to specify the number of partitions but automatically divides them through the calculation process. The disadvantage is that it is challenging to process high-dimensional data. When the datasets' density is uneven and the cluster spacing changes significantly, the clustering

impact is low. The advantage of the CURE clustering algorithm is that it can find clusters in complex spaces and is less affected by noise. The disadvantage is that there are many parameters, including sampling size, number of clustering, shrinkage ratio, etc. There is a sampling error. It is challenging to find spatial clusters with very complex shapes (such as hollow shapes) sensitive to spatial data density differences. Although THE CURE clustering algorithm is designed for large databases, the efficiency is still insufficient when the data volume increases rapidly.

3. Experiments

3.1. Common Data Sets

Standard clustering analysis datasets include clustering and classification datasets, and common indicators of datasets include record count, feature number, category, simple distribution and other indicators. The details are as follows:

Table 1. Cluster dataset information

datasets	record count	feature number	category	simple distribution	overlap
iris	150	4	3	50/50/50	No
wine	178	13	3	59/71/48	No
emotions(music)	593	72	6	173/166/264/148/168/189	YES
yeast	2417	103	14	mixed distribution	YES
scene	2407	294	6	427/364/397/433/533/431	YES
wdbc	569	30	2	212/357	No
breasttissue	106	9	6	21/15/18/16/14/22	No
seeds	210	7	3	70/70/70	No
glass	214	9	6(7)	70/76/17/13/9/29	No

Table 2. Classified dataset information

Datasets	sample size	characteristic number	category	simple distribution	missing value
appendicitis	106	7	2	21/85	No
balance	625	4	3	288/49/288	No
banana	5300	2	2	2924/2376	No
yeast	2417	103	14	Mixed distribution	YES
bands	365(539)	19	2	230/135	Yes
bupa	345	6	2	145/200	No

3.2. Evaluation Indicators

In order to judge the effect of clustering, we usually use the following evaluation indicators: Purity, Rand index (RI) and F-score.

3.2.1 Purity

Clustering purity is also known as the accuracy of clustering. It is the simplest and most intuitive method in the evaluation standard of clustering results. The calculation formula of purity is defined as follows:

$$P = (\Omega, C) = \frac{1}{N}\sum_{k} \max_{j} \left|\omega_k \cap C_j\right|$$

Where N represents the total number of samples; $\Omega = \{w_1, w_2, \cdots, w_k\}$ represents clusters after clustering, while $C = \{C_1, C_2, \cdots, C_J\}$ represents the correct category, ω_k represents all samples in

the k cluster after clustering, C_j represents the real sample in the j category. Here, the value range of P is $[0,1]$. The larger the value, the better the clustering effect.

3.2.2 Feature Representation

For Rand coefficient and F value, we first need to give the definitions that follow:

TP: stands for the quantity of two related sample points in a cluster.

FP: indicates the quantity of two different sample points in a cluster.

TN: indicates the quantity of two clusters with two different sample points.

FN: indicates the number of comparable sample points in two clusters.

With the above four definitions, we can define the calculation formula of Rand coefficient[12], F value[13], Precision[14] and Recall[15]:

$$RI = \frac{TP+TN}{TP+FP+FN+TN} \tag{1}$$

$$Precision = \frac{TP}{TP+FP} \tag{2}$$

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

$$F_\beta = (1+\beta^2)\frac{Precision \cdot Recall}{\beta^2 \cdot Precision+Recall} \tag{4}$$

Here, RI and F_β, the range of value is $[0,1]$, and the greater it is, the better the clustering effect is.

3.3. Performance Comparison

We select and compare the representative algorithms, and the five clustering methods are summarized in the Table 3 below.

Table 3. Performance of different algorithms[16]

Category	Algorithm	Time complexity	Scalability	Suitable for large scale data	Suitable for high dimensional data	Sensitive of noise
Partition	k-means	Low O(knt)	Middle	Yes	No	High
	K-medoids	High O(k(n−k)ˆ2))	Low	No	No	Little
Hierarchy	CLARANS	High O(nˆ2)	Middle	Yes	No	Little
	BIRCH	Low O(n)	High	Yes	No	Little
	CURE	Low O(s ˆ2*logs)	High	Yes	Yes	Little

4. Discussion and Conclusion

Considering the ongoing advancement of deep learning algorithms, clustering algorithms for convolutional neural networks-based clustering have made some progress, have become an effective model for the combination of deep learning and machine learning, and have achieved relatively good performance on many public datasets. Therefore, the subsequent research can start from the following aspects:

(1) Data preprocessing. The data problem must be resolved as the initial step in any clustering algorithm. As a result of different application problems, different models need to be used, and it is imperative whether data suitable for clustering can be constructed through data preprocessing or throughout the feature extraction procedure.

(2) The underlying issue of high computational complexity and inadequate dimensionality reduction impact in the clustering process still exists with the enormous development of information, which leads to the decline of the clustering effect. In the future, we can explore how to rely on unsupervised, semi-supervised, self-supervised and other methods to help neural networks perform repeated optimization, such as using comparative learning to represent data better or in complex problems with the assistance of well-designed model architectures—disassembly of the clustering task.

(3) In the deep clustering of GNN, how to fully exploit the graph details and the properties and characteristics of the graph structure still lacks a reasonable and holistic solution or idea. The problem of fully extracting information from large-scale or cross-directional datasets can be solved by building a knowledge graph.

(4) Graph data structure is very different, with heterogeneity and dynamics. Therefore, how to solve the problem of graph heterogeneity and dynamics according to the critical subgraph structure is a direction worth exploring.

(5) The clustering methodology is used for discovering the intrinsic between data structure and is a fundamental problem in many data-driven application fields. This paper summarizes the research status of methodologies for clustering in recent decades. Specifically, the relevant critical steps of the clustering algorithm are first introduced. The representative clustering algorithms are further introduced in detail from the two aspects of partition and hierarchical clustering, and the common problems and application directions in the research process are summarized.

References

[1] Han J, Kamber M, Pei J: 1 - Introduction, Han J, Kamber M, Pei J, editor, Data Mining (Third Edition), Boston: Morgan Kaufmann, 2012: 1-38.

[2] Lam D, Wunsch D C. Clustering [J]. Academic Press Library in Signal Processing, 2014, 1: 1115-1149.

[3] Nagpal A, Jatain A, Gaur D. Review based on data clustering algorithms[C]. 2013 IEEE conference on information & communication technologies, 2013: 298-303.

[4] Macqueen J. Classification and analysis of multivariate observations[C]. 5th Berkeley Symp. Math. Statist. Probability, 1967: 281-297.

[5] Ng R T, Han J. CLARANS: A method for clustering objects for spatial data mining [J]. IEEE transactions on knowledge and data engineering, 2002, 14(5): 1003-1016.

[6] Fix E, Hodges J L. Discriminatory analysis. Nonparametric discrimination: Consistency properties [J]. International Statistical Review/Revue Internationale de Statistique, 1989, 57(3): 238-247.

[7] Piryonesi S M, El-Diraby T. Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition [J]. Journal of Infrastructure Systems, 2020, 26.

[8] Kostinakis K, Morfidis K, Demertzis K, et al. Classification of Buildings' Potential for Seismic Damage by Means of Artificial Intelligence Techniques[J]. arXiv preprint arXiv:2205.01076, 2022.

[9] Boshnakov G N. Introduction to Time Series Analysis and Forecasting, Wiley Series in Probability and Statistics, by Douglas C. Montgomery, Cheryl L. Jennings and Murat Kulahci (eds). Published by John Wiley and Sons, Hoboken, NJ, USA, 2015. Total number of pages: 672 Hardcover: ISBN: 978-1-118-74511-3, ebook: ISBN: 978-1-118-74515-1, etext: ISBN: 978-1-118-74495-6: Wiley Online Library, 2016.

[10] Murtagh F. A survey of recent advances in hierarchical clustering algorithms[J]. The computer journal, 1983, 26(4): 354-359.

[11] Ester M, Kriegel H-P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]. kdd, 1996: 226-231.

[12] Rand W M. Objective Criteria for the Evaluation of Clustering Methods[J]. Journal of the American Statistical Association, 1971, 66(336): 846-850.

[13] Saxena A, Prasad M, Gupta A, et al. A review of clustering techniques and developments[J]. Neurocomputing, 2017, 267: 664-681.

[14] Olson D L, Delen D. Advanced data mining techniques[M]. Springer Science & Business Media, 2008.

[15] Powers D M. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation[J]. arXiv preprint arXiv:2010.16061, 2020.

[16] Xu D, Tian Y. A comprehensive survey of clustering algorithms[J]. Annals of Data Science, 2015, 2(2): 165-193.