International Journal of Data Warehousing and Mining
Volume 19 • Issue 1

Constrained Density Peak Clustering

Viet-Thang Vu, Faculty of Information Technology, Hanoi University of Industry, Vietnam*
T. T. Quyen Bui, Institute of Information Technology, Vietnam Academy of Science and Technology, Vietnam
Tien Loi Nguyen, Center of Information and Technology, Hanoi University of Industry, Vietnam
Doan-Vinh Tran, VNU University of Education, Vietnam National University, Hanoi, Vietnam
Hong-Quan Do, FPT University, Hanoi, Vietnam
Viet-Vu Vu, CMC University, Hanoi, Vietnam
Sergey M. Avdoshin, HSE University, Russia

ABSTRACT

Clustering is a commonly used tool for discovering knowledge in data mining. Density peak clustering (DPC) has recently gained attention for its ability to detect clusters with various shapes and noise, using just one parameter. DPC has shown advantages over other methods, such as DBSCAN and K-means, but it struggles with datasets that have both high and low-density clusters. To overcome this limitation, the paper introduces a new semi-supervised DPC method that improves clustering results with a small set of constraints expressed as must-link and cannot-link. The proposed method combines constraints and a k-nearest neighbor graph to filter out peaks and find the center for each cluster. Constraints are also used to support label assignment during the clustering procedure. The efficacy of this method is demonstrated through experiments on well-known data sets from UCI and benchmarked against contemporary semi-supervised clustering techniques.

KEYWORDS

AI, Data Mining, Machine Learning, Soft Computing

INTRODUCTION

The goal of clustering is to group a collection of objects together in a way that maximizes similarity within a cluster and dissimilarity between clusters (Ezugwu et al., 2022; Xu & Wunsch, 2005). It is a popular machine learning technique used in various fields, such as image processing, text mining, social science, and big data analysis, to mention just a few (Ezugwu et al., 2022; Krishnaswamy et al., 2023; Saha & Mukherjee, 2021; Chen et al., 2022; Liang & Chan, 2021; Hoi et al., 2022). Clustering can reveal the underlying structure of data, identify relationships between objects, and even detect

DOI: 10.4018/IJDWM.328776 *Corresponding Author

outliers. Since it is an unsupervised learning task, clustering does not rely on prior data knowledge,

Nevertheless, recent advances in machine learning have given rise to semisupervised clustering as a promising research area. Semisupervised clustering algorithms can leverage side information, such as labeled data or constraints, to enhance clustering quality and efficiency (Basu et al., 2008).

According to Jonschkowski et al. (2015), side information refers to additional data that are not part of the input or output space but can be helpful in the learning process. It is also used in other machine learning models, including support vector machines, multiview learning, and deep learning (Jonschkowski et al., 2015; Geoffrey et al., 2011). Generally speaking, side information can be expressed as constraints or labeled data, also known as seeds. In this paper, the following constraints are used to guide the clustering process for a given data set $X = \{x_1, x_2, \ldots, x_n\}$:

Must-Link: A Must-Link constraint between two data points x_i and x_j indicates that they must be placed in the same cluster.

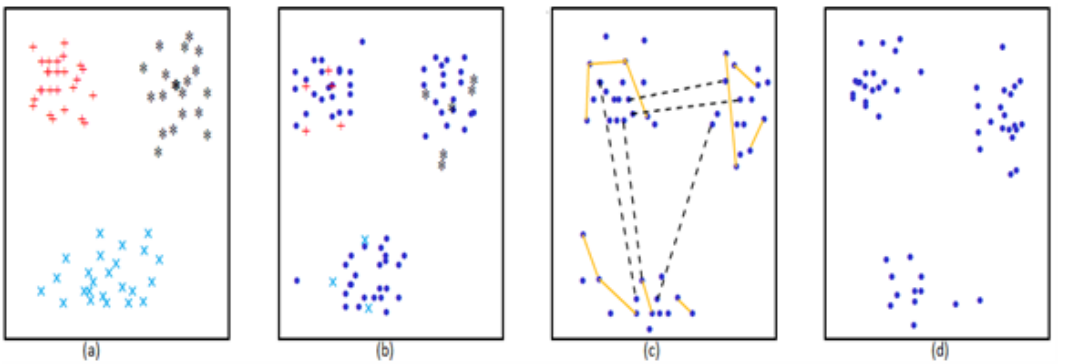
Cannot-Link: A Cannot-Link constraint between two data points x_i and x_j indicates that they must be placed in separate clusters and should not be grouped together.

In Figure 1, a graphical illustration of the various types of side information that can be incorporated for data classification is presented. Over the last 20 years, several semisupervised clustering techniques have been developed in the literature. Typically, these methods are derived from unsupervised algorithms and aim to incorporate side information to improve clustering performance. Some of the significant techniques in this category include semisupervised K-means (Pelleg & Baras, 2007; Basu et al., 2002, 2004; Bilenko et al., 2004; Davidson & Ravi, 2005), semisupervised fuzzy clustering (Bensaid et al., 1996; Maraziotis, 2012; Abin, 2016; Grira et al., 2008), semisupervised spectral clustering that has been investigated in various research papers (Mavroeidis, 2010; Wang et al., 2014; Mavroeidis & Bingham, 2010), semisupervised density-based clustering (Böhm & Plant, 2008; Lelis & Sander, 2009; Ruiz et al., 2010; Vu et al., 2019), semisupervised hierarchical clustering that has been explored in Davidson and Ravi (2009), and semisupervised graph-based clustering (Kulis et al., 2009; Anand & Reddy, 2011; Vu, 2018), to mention a few.

Density peak clustering (DPC) was first proposed in Rodriguez & Laio (2014) and has since attracted considerable attention in Mehmood et al. (2016); Lin (2019, 2021); Chen et al. (2020); Zhou et al. (2018); Sieranoja and Fränti (2019); and Wang et al. (2020). DPC stands out by its ability to detect clusters of arbitrary shapes while maintaining robustness to noise (Vu et al., 2022; Xie et al., 2016), making it superior to traditional clustering methods, such as DBSCAN and K-means. Nonetheless, recent studies have indicated that DPC struggles when clusters have different densities or local peaks (Ding et al., 2020b; Lin et al., 2020).

Figure 1. The four categories of machine learning: supervised (a), labeled (b), constrained (c), and unsupervised (d)

Note. The data points that have not been labeled are represented by dots, while circles, asterisks, and crosses are used to represent labeled points. In the case of constrained learning (c), solid and dashed lines are used to indicate the Must-Link and Cannot-Link constraints, respectively (Lange et al., 2005).



Hence, our aim is to present a novel semisupervised density peak clustering method, called density peak clustering with constraints (CDPCs), that addresses this limitation and demonstrates commendable performance on data sets with varying densities. The key contributions of this work can be outlined as follows:

To identify peaks in the clustering process, our method employs a k-nearest neighbor graph and connected components constructed using the concept of a strong path. The distance between vertices in the k-nearest neighbor (k-NN) graph is determined by shared nearest neighbor points, making it unaffected by data density.

We efficiently embed constraint sets in the clustering process. Our approach utilizes constraints not only in the k-NN graph but also in the label propagation to form clusters.

Based on these key ideas, we develop a novel method called CDPCs, which overcomes the main limitations of DPC. As far as we are aware, CDPCs is the initial semisupervised density peak clustering approach presented in the literature.

We performed thorough experiments on UCI and real-world data sets to showcase the efficacy of our novel approach in comparison to contemporary semisupervised clustering techniques.

The subsequent sections of this article are structured as follows. Related Works offers an overview of existing literature on semisupervised clustering techniques. Density Peaks Clustering With Constraints is dedicated to the presentation of our innovative technique for semisupervised density peak clustering with constraints, named CDPCs. Experimental Results details our experimental results, and the Conclusion concludes the paper, outlining potential avenues for future research.

RELATED WORKS

Over the last two decades, there has been considerable interest in semisupervised clustering, a field that employs supplementary data to enhance the quality of clustering. By utilizing a small set of side information, clustering can be both improved and expedited (Basu et al., 2008). Currently, there exist four approaches for integrating side information, namely, enforcing, penalty-based, metric learning, and declarative methods.

To begin with, incorporating side information directly into the clustering process can be achieved using the enforcing technique. This technique involves incorporating the side information in various ways, such as conditioning data point allocation to clusters, initializing the centroids at the start of the K-means algorithm, employing voting clustering algorithms, and so forth. For example, one of the earliest semisupervised clustering algorithms (Wagstaff et al., 2001) integrated constraints into the K-means clustering assignment loop, resulting in the COP-K-means method outperforming the K-means algorithm. Another approach, presented in Basu et al. (2002), is a seed-based clustering technique, where labeled data are used to calculate k centers in the first phase of K-means. Despite its straightforward approach, the clustering results produced by seed K-means were reliable and effective.

Additionally, the authors in Ruiz et al. (2010) introduced the constraint DBSCAN (C-DBSCAN) technique. Initially, C-DBSCAN leverages a KD-Tree to divide the data space into denser subspaces and generate a collection of initial local clusters. If KD-leaf tree nodes contain specific point groups, they are separated to satisfy the Cannot-Link constraints associated with their contents. Following that, density-based local clusters are merged by implementing Must-Link constraints. Finally, a bottom-up approach is used to integrate the adjacent neighborhoods while applying the remaining Cannot-Link constraints.

MCSSDBS, presented in Vu et al. (2019), is a significant development in the field of semisupervised density-based clustering, as it was the first algorithm designed to incorporate constraints and seeds into the clustering process. During the nearest-neighbors identification process, MCSSDBS uses Must-Link and Cannot-Link constraints while constructing a minimum spanning tree (MST) for a complete graph. The algorithm then enlarges the clusters from the seeds, but the authors

hypothesized that the longest edge may not be a suitable cut point. Thereby, it uses an active learning process to obtain a label from users for the pair (p_i, p_{i+1}) and employs Cannot-Link constraints. If no information is available, the largest value (p_i, p_{i+1}) of the MST is utilized. In the final stage, Must-Link constraints are employed again to combine isolated points that are connected to clusters, and the remaining unassigned points are treated as outliers.

Additionally, Vu (2018) introduced a clustering approach based on graphs and seed points. The method represents the data set using a k-NN graph and uses seeds in the partitioning stage to create connected components, also known as principal clusters. Each connected component includes only one type of seed. This approach is advantageous, as having seeds makes it easier to find the optimal solution for partitioning the graph into clusters. By following this method, constraints were enforced during the clustering process, leading to improved results compared to SSDBSCAN.

Another semisupervised clustering technique, agglomerative hierarchical clustering with constraints (AHCCs), was introduced in Davidson and Ravi (2009). In AHCCs, constraints were embedded in the distance matrix between every pair of points, and the shortest path strategy was used to update the matrix distance. AHCCs is considered a practical method for semisupervised clustering in literature.

The second strategy, referred to as penalty-based, recognizes that certain constraints may not be satisfied during the clustering procedure. This is known as the soft constraints embedding process. Several notable algorithms have been proposed in this approach. An algorithm called constrained vector quantization error (CVQE) was introduced in Davidson and Ravi (2005) as a means of addressing the limitations of existing clustering methods. CVQE is a semisupervised clustering algorithm that is built upon the K-means algorithm. The algorithm begins by initializing the cluster centroids according to the constraints and then iteratively refines them according to the traditional VQE process. The main idea behind CVQE is to help guide the formation of clusters to satisfy certain requirements while still maintaining the flexibility of unsupervised learning. This approach was seen as innovative and has since been improved with methods such as LCVQE (Pelleg & Baras, 2007) and CVQE+ (Mai et al., 2018). These improvements have further enhanced the algorithm's performance, making it a promising approach for clustering tasks.

In Basu et al. (2004), another variation of the K-means algorithm is proposed, which includes constraints in the clustering process by giving weights to Must-Link and Cannot-Link constraints. During the assignment process, the algorithm aims to ulfil as many constraints as possible to ensure that the resulting clusters adhere to the given constraints.

In addition, AFCC, a constrained fuzzy C-means method, was introduced in Grira et al. (2008). This algorithm embeds constraints in the membership matrix by utilizing an objective function. The penalty for violating Must-Link and Cannot-Link constraints is adjusted based on the membership values, which help ensure that the given constraints are satisfied to the greatest extent possible. Furthermore, Antoine et al. (2012) proposed the constraints evidential C-means (CECMs) algorithm, which includes a penalty term in the objective function that considers the provided constraints.

The metric-based learning approach is the third strategy employed for integrating constraints in clustering. This technique trains a metric using constraints to ensure that similar instances (as specified by the Must-Link constraints) are placed closer to each other while dissimilar instances (as specified by the Cannot-Link constraints) are placed further apart. Several distance metrics have been used for this purpose. One such metric is the Jensen-Shannon divergence, which can be trained using gradient descent, as described in Cohn et al. (2003).

Another approach involves modifying the traditional Euclidean distance by incorporating a shortest-path algorithm, as discussed in Klein et al. (2002). Additionally, Mahalanobis distances, which can be trained using convex optimization techniques, have also been employed, as described in Eric et al. (2002); Bar-Hillel et al. (2003).

Other methods include distance metric learning based on Discriminative Component Analysis (MDCA) in Steven et al. (2006) and a nonlinear metric learning method that learns a completely flexible distance metric via learning a nonparametric kernel matrix in Baghshah and Shouraki (2010). Another example of a semisupervised K-means clustering method that integrates enforcement and metric learning into a single framework is MPCK-means (Bilenko et al., 2004).

Finally, declarative methods offer a mathematical approach to solve clustering problems using models like integer linear programming (ILP), SAT, and constraint programming. ILP-HC was proposed in Gilpin and Davidson (2017), which is an ILP-based method for hierarchical constraint clustering. It enforces dendrogram features as linear constraints and uses high-quality solvers like CPLEX and Gurobi to find solutions and utilize multicore architectures. In Dao et al. (2017), a comprehensive framework for constrained clustering was introduced that is declarative and based on constraint programming. The suggested method makes use of several optimization criteria, including diameter, split, within-cluster sum of dissimilarities, and within-cluster sum of squares, as well as different types of user constraints, to address diverse constrained clustering problems. Moreover, a two-cluster problem was introduced in Davidson et al. (2010). The authors have created an effective clustering method by transforming both instance-level constraints, such as Must-Link and Cannot-Link, and cluster-level constraints, such as maximum diameter and minimum separation, into a 2-satisfiability (2SAT) problem. Overall, these newly proposed algorithms represent a novel research direction that requires further exploration in recent years.

DENSITY PEAKS CLUSTERING WITH CONSTRAINTS

Density Peak Clustering

Density peak clustering is a clustering method that utilizes the concept of density estimation. In this method, given a data set $X = \{x_1, x_2, \ldots, x_n\}$, the local density of a point *I*, represented by ρ_i, is computed using Equation (1):

$$p_i = \sum_j x\left(d_{ij} - d_c\right), \tag{1}$$

where d_{ij} is the distance between the point *i* and point *j*, and d_c is a cutoff distance. $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise. The δ_i value, which is measured by computing the minimum distance between the point *i* and any other point with a higher density, but, for the point with the highest density, δ_i is set to $\max_j(d_{ij})$. It is defined in Equation (2).

$$\delta_i = \begin{cases} \min\limits_{j:p_j > p_i}\left(d_{ij}\right), & if\ \exists j\ subject\ to\ p_j > p_i \\ \max\limits_{j}\left(d_{ij}\right), & otherwise \end{cases}. \tag{2}$$

The decision graph is constructed based on ρ_i and δ_i. Figure 2 illustrates an example of a decision graph, where we can select the points located in the top-right corner as the cluster centers, and subsequently assign each point in the data set to the cluster of its nearest neighbor with a higher density. DPC has proven to be effective when compared to traditional clustering techniques, such as K-means and DBSCAN. Nevertheless, DPC has a significant drawback when applied to data sets where the density varies across different clusters. Figure 3 provides an example where DPC fails to identify the correct peaks for two clusters.

Figure 2. A data set (left) and its decision graph (right) (Rodriguez & Laio, 2014)

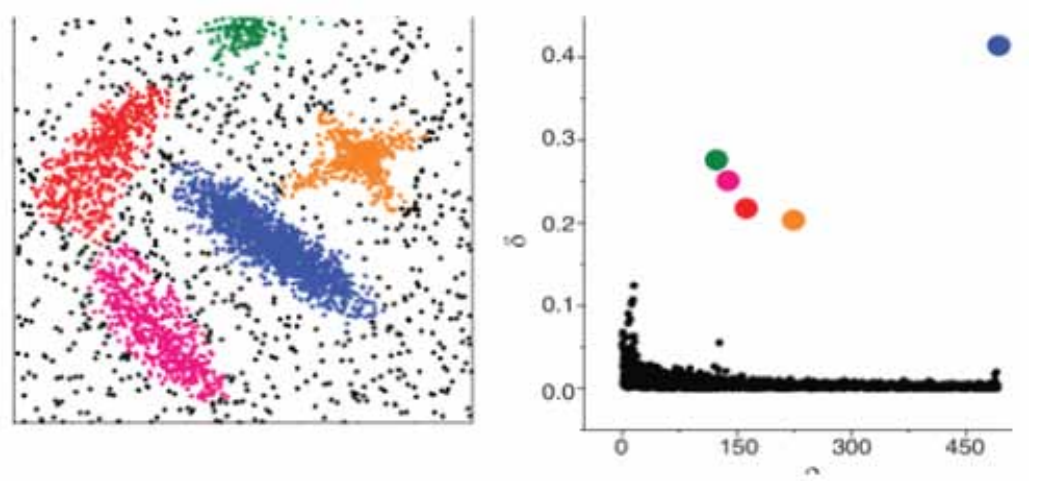
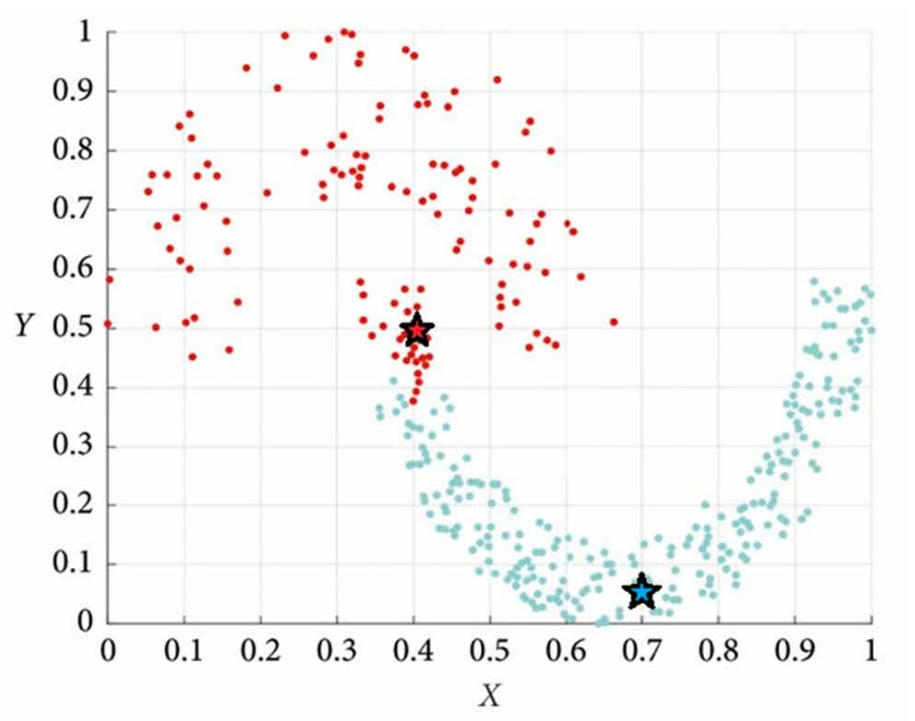


Figure 3. Clustering results of DPC on the Jain data set (Ding et al., 2020a)

Note. The stars mark improper peaks found by the DPC algorithm.



The k-Nearest Neighbor Graph

The k-NN graph is a graph-based machine learning algorithm used for classification and clustering tasks. To create the k-NN graph for a data set X consisting of n data points, denoted as $\{x_1, x_2, \ldots, x_n\}$, one must calculate the distance between each pair of data points using a specified distance metric, for

example, the Euclidean distance. Subsequently, for each point, the K-nearest neighbors are identified based on the computed distances, and edges are formed connecting the point to its nearest neighbors.

The weight of two vertices in the k-NN graph can be computed using different approaches. Given NN(.) is the set of k-nearest neighbors for a specified point, one prevalent approach (Jarvis & Patrick, 1973) is to use the number of shared nearest neighbors between the two points, denoted as $\left|NN\left(x_i\right) \cap NN\left(x_j\right)\right|$, and the total number of neighbors of the two points, denoted as $\left|NN\left(x_i\right) \cup NN\left(x_j\right)\right|$. The weight can be calculated using Equation (3), also named as structural Jaccard similarity:

$$\omega\left(x_i, x_j\right) = \frac{\left|NN\left(x_i\right) \cap NN\left(x_j\right)\right|}{\left|NN\left(x_i\right) \cup NN\left(x_j\right)\right|}. \tag{3}$$

The Proposed Method CDPCs

As previously mentioned, one of the limitations of DPC is its inability to effectively handle data sets with varying densities. In fact, each cluster may contain some local peaks, so it needs to be corrected in the peak finding process. To address this issue, we introduce a new semisupervised graph-based clustering algorithm called CDPCs (an abbreviation for **C**onstrained **D**ensity **P**eak **C**lustering), which incorporates constraints in peak finding. The main concept is to use a k-NN graph to represent data and a set of constraints to improve the peak finding process. In cases where peaks appear in the same region, we use the concept of a strong path proposed by Vu et al. (2012) to identify and label them as the same region. The definition of a strong path in a k-NN graph is given in Definition 1, where two vertices *u* and *v* are in the same region of density if there exists a strong path between them. It should be noted that the distance between *u* and *v* is well-suited for clusters with varying densities when utilizing the aforementioned k-NN graph. Our CDPCs algorithm utilizes the strong path for peak propagation, merging peaks that share a strong path in the peaks filtering process. This approach is inspired by the density-based clustering concept, where clusters are defined as high-density regions separated by low-density regions.

Definition 1: *(Adopted from (Vu et al., 2012)) Given a k-nearest neighbor graph (k-NNG) for a data set X, a threshold θ, and a set of constraints Y (Y can be empty), a path from vertex u to vertex v is defined as a strong path* $SP\left(u, v, \theta\right)$ *if there exists a sequence of vertices* $\left(z_1, z_2, \ldots, z_t\right)$, *such that* $u = z_1$, $v = z_t$ *and* $\forall i = 1 \ldots t-1 : \omega\left(z_i, z_{i+1}\right) \geq \theta$ *or* $z_i, z_i + 1$ *is a Must-Link constraint in Y.*

The CDPC's primary steps are displayed in Figure 4 and can be summarized as follows:

Step 1: Constructing a k-NN graph. This stage involves creating a k-NN graph that employs constraints to determine the weights of the graph. The Must-Link and Cannot-Link constraints are used in this process.

Step 2: Embed constraints to identify connected components. From the k-NN graph, we can identify all connected components, which are then used in the next step of peak refinement (Steps 3–9). Peaks within the same connected component are merged using the definition of the strong path.

Step 3: Computing ρ, δ, and identifying final peaks. Like the DPC, in this step, ρ and δ are calculated and used to generate the decision graph, which is then employed to identify the final peaks. A loop is utilized to find the peak set from the decision graph. Initially, the first peak is added, and at each iteration *t*, a new peak will be added if there are no strong paths connected to the current set of peaks. The loop is stopped by the users when they determine that the collected peaks are sufficient for the clustering process.

Step 4: Clustering process. Once the final peaks are identified, the original DPC method is employed to detect clusters. During the label propagation step, unlabeled points receive the same label as the point with the highest density, provided that it adheres to the Cannot-Link constraints. Importantly, constraints can prevent errors that arise from this step. For example, in DPC, high-density points may be assigned to the nearest neighbor point with higher density, but these points may actually belong to different clusters. In these cases, the Cannot-Link constraints can prevent misclassification and accurately identify the correct peak for these unlabeled points.

Algorithm 1 outlines the proposed CDPCs method, which involves constructing a k-NN graph with a complexity of $O(n^2)$ or $O(n \times \log(n))$ for low dimension data as stated in Ertoez et al. (2003). Step 2 of the algorithm, which involves extracting the connected components, has a complexity of $O(n \times k)$, where k refers to the number of neighbors. Furthermore, the DPC has a complexity of $O(n^2)$, resulting in an overall complexity of $O(n^2)$ for the CDPCs method.

EXPERIMENTAL RESULTS

Data Sets

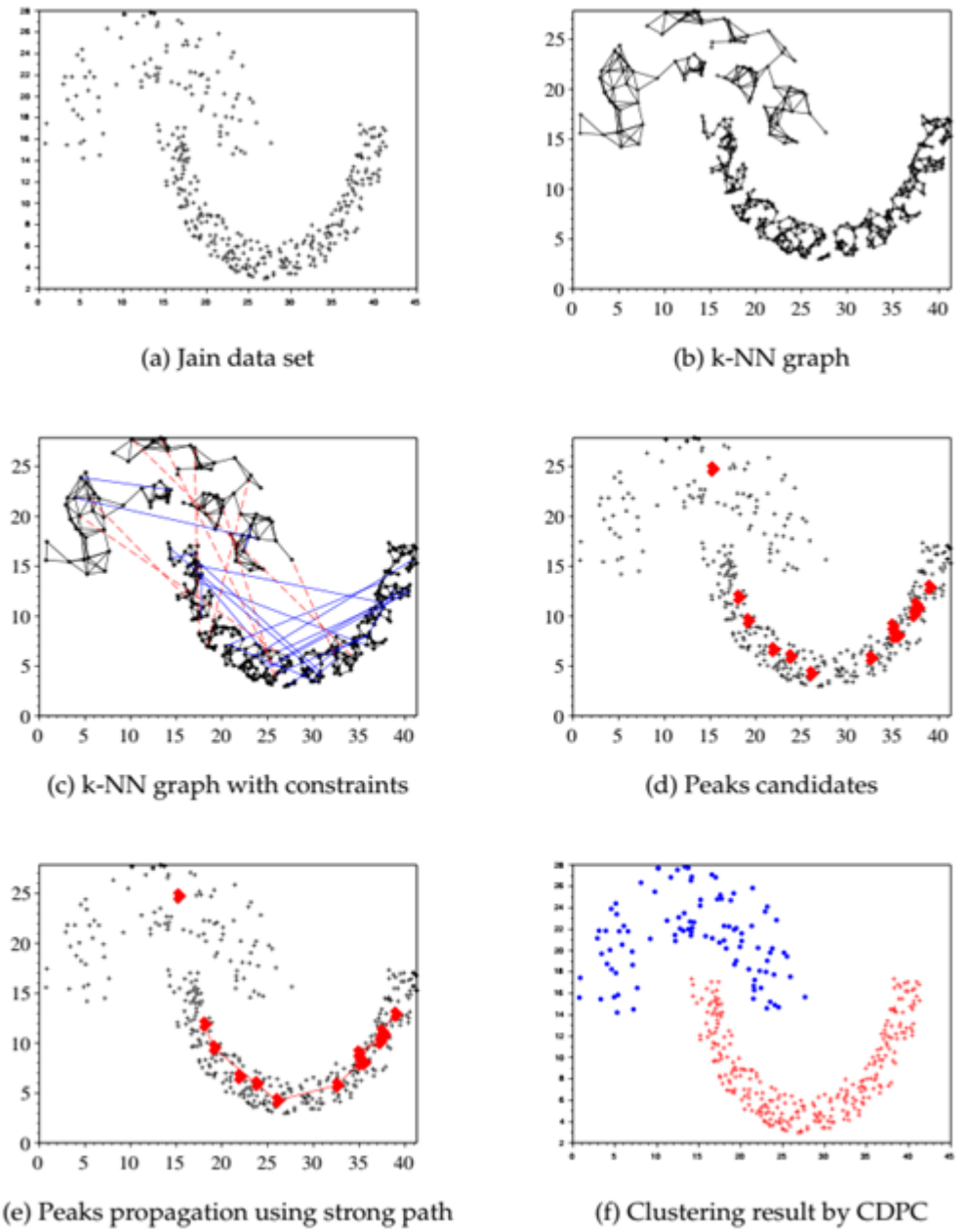
Nine commonly used data sets from the Machine Learning Repository (Asuncion & Newman, 2015) and one data set extracted from AWID intrusion detection data (Kolias et al., 2015) were utilized in this study. Table 1 presents the characteristics of these data sets. It is worth mentioning that these

Algorithm 1. CDPC algorithm

Algorithm 1 CDPC Algorithm

Input: Data set $\mathcal{X}$, number of neighbors k, a set of constraints $\mathcal{C}$, d
Output: A partitioning of $\mathcal{X}$
Process:
1: Construct the k-NN graph for $\mathcal{X}$
2: Embed constraints to obtain connected components
3: **for all** *connected components* having at cannot-link constraints **do**
4: $t = 1$
5: **repeat**
6: All edges whose weights equal to t will be deleted.
7: $t = t + 1$
8: **until** Cannot-link violation = false
9: **end for**
10: Calculating $\rho_i, \forall i = 1..n$;
11: Calculating $\delta_i, \forall i = 1..n$;
12: Constructing the decision graph;
13: **repeat**
14: Finding the peak p_j such that $\rho_j \cdot \delta_j$ is being maximum;
15: Refinement of the peaks using the connected components;
16: Update the decision graph;
17: **until** user_stop = true
18: Running DPC (using constraints in the label propagation step) on the collected peaks to obtain clusters of X.
19: Return the results of partitioning of $\mathcal{X}$.

Figure 4 Illustration of our proposed method: CDPCs



data sets were chosen due to their reputation as standard data sets for evaluating clustering algorithm performance (Basu et al., 2008; Lelis & Sander, 2009; Abin & Vu, 2020).

Furthermore, in order to demonstrate the suitability of our proposed method across different data sets, we have intentionally selected a variety of data sets. These data sets encompass not only those with uniform density but also instances that exhibit varying density levels. For instance, as illustrated in Figure 5, from the decision graph, it is challenging to determine the cluster centers (peaks) for

Table 1. Details of the data sets utilized in experiments

ID	Data	#Objects	#Attributes	#Clusters
1	Iris	150	4	3
2	Glass	214	9	6
3	Protein	115	20	6
4	Soybean	47	35	4
5	Breast	568	30	2
6	Image	2,100	19	7
7	Wine	178	13	3
8	Balance	625	4	3
9	Sonar	208	60	2
10	AWID3	12,000	35	2

data sets such as Iris, Glass, and Sonar. This indicates that these data sets lack uniform density or may have overlapping clusters.

Evaluation Method

The Rand Index (RI) serves as a measure of the similarity between two clustering results. It is a popular measure of clustering performance due to its straightforward calculation and interpretation. Given two sets of data points A and B, the Rand Index calculates the quantity of data point pairs that are categorized as either true positives (in the same cluster in both sets) or true negatives (in different clusters in both sets). This value is subsequently divided by the total number of possible data point pairs.

It is computed in Equation (4) as follows:

$$RI = \frac{TP + TN}{TP + FP + FN + TN}, \tag{4}$$

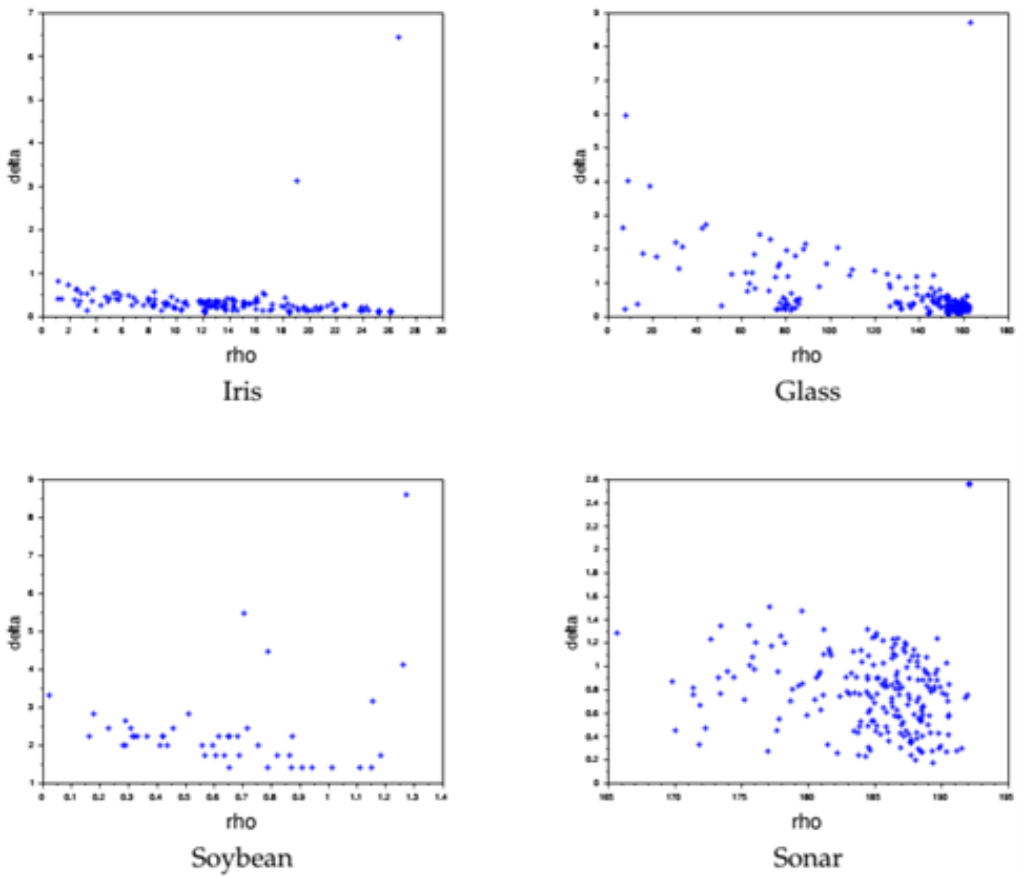
where:TP (true positives) refers to the quantity of data point pairs that belong to the same cluster in both sets.TN (true negatives) represents the number of data point pairs that are categorized as belonging to different clusters in both sets.FP (false positives) refers to the number of data point pairs that are categorized as belonging to the same cluster in set A but are deemed to be in different clusters in set B.FN (false negatives) denotes the number of data point pairs that are categorized as belonging to different clusters in set A but are deemed to be in the same cluster in set B.

The Rand Index varies between 0 and 1, where a higher value of the RI indicates a higher degree of agreement between the two sets of clustering results. An RI of 1 signifies complete agreement between the two sets of data points. Conversely, an RI of 0 means that the clustering results in both sets are completely different. In our experiments, we reported the RI as a percentage.

Comparative Results

To demonstrate the effectiveness of our suggested CDPCs algorithm, we conducted a comparison study with two well-known constrained clustering algorithms, MPCK-means and AHCC. Additionally, we compared the results of CDPCs with those obtained by the original DPC algorithm. Randomly generated constraints were utilized in our experimentation, and the results were averaged over 50

Figure 5. The decision graph for iris, glass, soybean, and sonar



runs. The k value is set to 4 for Iris, Glass, and Breast; 5 for Protein, Soybean, Breast, and Image; 6 for Wine, Balance, and Sonar; and 8 for AWID3.

While Figure 5 provides a decision graph that explains the peaks finding phase, RI results obtained by the algorithms are displayed in Figure 6. Upon observing these graphs, it is apparent that CDPCs performed better than AHCCs and MPCK-means in most cases. Although MPCK-means and AHCCs outperformed the original DPC algorithm, CDPCs significantly enhanced the clustering performance. This improvement by CDPCs can be attributed to the fact that DPC cannot correctly identify the appropriate peaks for each cluster, as indicated by recent research (Ding et al., 2020b; Lin et al., 2020). CDPCs, on the other hand, employ constraints and a k-NN graph to identify the correct peaks for clusters, even when the density of clusters differs.

The proposed method, CDPCs, shows significant performance improvements compared to DPC. This is particularly evident in the Iris data set, which consists of three clusters, with two clusters exhibiting considerable overlap, as shown in Figure 5. While DPC only identifies two peaks in this situation, CDPCs successfully detect three peaks by incorporating constraints and a k-NN graph, leading to an optimal outcome. Similarly, for the Glass data set, as shown in Figure 5, DPC struggles to accurately identify the number of peaks. Conversely, CDPCs achieve better results by utilizing 50 constraints.

The Soybean data set, which has four clusters, was also analyzed using a decision graph in Figure 5. Following the DPC condition led to a mistake, as the third and fourth identified peaks belonged to

the same cluster. However, CDPCs were able to identify the correct peaks for each cluster after the peak-filtering process mentioned earlier, resulting in the detection of four peaks and demonstrating a good performance.

Moreover, in the Protein, Breast, and Image data sets, CDPCs showcase slightly superior outcomes than MPCK-means and AHCCs and notably outperform DPC. Additionally, Figure 6 illustrates that CDPCs surpass AHCC, MPCK-means, and DPC on Wine, Balance, Sonar, and AWID3 data sets. For instance, in the Sonar data set consisting of two clusters, the decision graph in Figure 5 demonstrates that the DPC algorithm can only identify one peak candidate. In contrast, CDPCs enhance the performance by 5% compared to DPC.

Overall, by incorporating constraints and a k-NN graph into the peak-finding process, CDPCs effectively overcome the main limitation of DPC and successfully identify suitable peaks for clustering purposes.

Selection of Constraint Sets

In CDPCs, the Must-Link and Cannot-Link constraints are utilized to support the peak finding and clustering process. However, in our current work, the constraint sets are generated randomly, which may result in the inclusion of noisy constraints. Consequently, the overall performance of the algorithm for each run relies on the quality of the constraint set. To tackle this challenge, there are generally two common approaches (Vu et al., 2012; Basu et al., 2004; Mallapragada et al., 2008; Grira et al., 2008). The first approach leverages domain knowledge and prior information about the data to identify the relevant constraints. Another approach is to incorporate optimization methods in order to search for the optimal constraint sets. This involves defining an objective function that quantifies the quality of the resulting clusters and then searching for the constraint sets that either maximize or minimize this objective function. However, it is important to acknowledge that the selection of constraint sets is still an ongoing research topic. As part of our future work, we plan to expand our research to tackle this issue. We aim to investigate and propose strategies for selecting constraint sets that are more effective.

CONCLUSION

The CDPCs, a novel semisupervised density peak clustering with constraints, was introduced in this paper. It addresses a significant limitation of DPC, specifically when clustering in varying densities. The proposed method utilizes constraints and a k-NN graph to accurately identify peaks. By using the concept of strong path and constraints, the peaks are merged or left alone depending on the propagation step, allowing for the detection of peaks in each region as a cluster. Experimental results on various data sets indicated that the CDPCs algorithm surpasses well-known semisupervised clustering algorithms in the literature, including MPCK-means and AHCC.

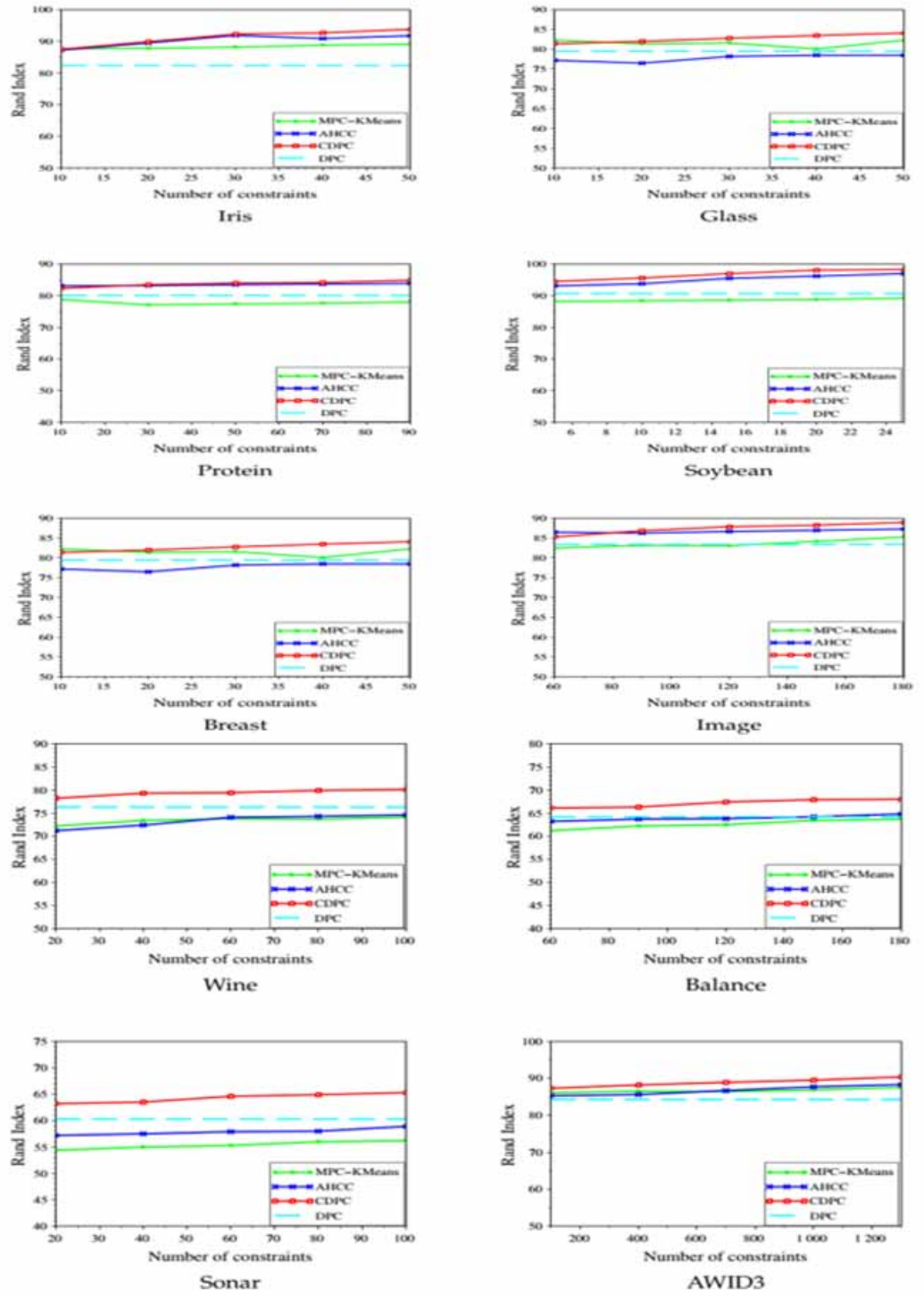
We intend to incorporate more supplementary side information, like seeds or obstacle/distance constraints in the upcoming work and tackle the issue of selecting appropriate constraints for density peaks clustering. Furthermore, it is essential to validate the proposed method in real-world applications. Additionally, it is essential to validate the proposed method in real-world applications. Our prior research (Vu et al., 2019) also delved into the examination of clustering methods utilizing data sets associated with facial expressions. Notably, data sets such as FER2013 (Goodfellow et al., 2013) exhibit significant diversity and offer abundant possibilities for evaluating the efficacy of clustering algorithms.

CONFLICT OF INTEREST

The authors of this publication declare there are no competing interests.

Figure 6. Comparison results on data sets among DPC, AHCC, MPCK-Means, and our CDPCs method

Note. The comparison is on data sets Iris, Glass, Protein, Soybean, Breast, Image, Wine, Balance, Sonar, and AWID3, respectively.



FUNDING INFORMATION

The reported study was funded by VAST (Vietnam Academy of Science and Technology) with project number QTRU01.14/21-22 in 2021.

REFERENCES

Abin, A. A. (2016). Clustering with side information: Further efforts to improve efficiency. *Pattern Recognition Letters*, *84*, 252–258. doi:10.1016/j.patrec.2016.10.013

Abin, A. A., & Vu, V.-V. (2020). A density-based approach for querying informative constraints for clustering. *Expert Systems with Applications*, *161*, 113690. doi:10.1016/j.eswa.2020.113690

Anand, R., & Reddy, C. K. (2011). Graph-based clustering with constraints. *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*. https://dmkd.cs.vt.edu/papers/PAKDD11.pdf

Antoine, V., Labroche, N., & Vu, V.-V. (2014). Evidential seed-based semi-supervised clustering. *Proceedings of the 2014 Joint 7th International Conference on Soft Computing and Intelligent Systems (SCIS) and 15th International Symposium on Advanced Intelligent Systems (ISIS)*. doi:10.1109/SCIS-ISIS.2014.7044676

Antoine, V., Quost, B., Masson, M.-H., & Denoeux, T. (2012). CECM: Constrained evidential C-means algorithm. *Computational Statistics & Data Analysis*, *56*(4), 894–914. doi:10.1016/j.csda.2010.09.021

Asuncion, A., & Newman, D. J. (2015). *UCI machine learning repository.* American Statistical Association. http://www.ics.uci.edu/~mlearn/MLReposit-ory.html

Baghshah, M. S., & Shouraki, S. B. (2010). Kernel-based metric learning for semi-supervised clustering. *Neurocomputing*, *73*(7–9), 1352–1361. doi:10.1016/j.neucom.2009.12.009

Bar-Hillel, A., Hertz, T., Shental, N., & Weinshall, D. (2003). Learning distance functions using equivalence relations. *Proceedings of the 20th International Conference on Machine Learning (ICML)*. https://dl.acm.org/doi/10.5555/3041838.3041840

Basu, S., Banerjee, A., & Mooney, R. J. (2002). Semi-supervised clustering by seeding. *Proceedings of the 19th International Conference on Machine Learning*. https://www.semanticscholar.org/paper/Semi-supervised-Clustering-by-Seeding-Basu-Banerjee/f4f3a10d96e0b6d134e7e347e1727b7438d4006f

Basu, S., Banerjee, A., & Mooney, R. J. (2004). Active semi-supervision for pairwise constrained clustering. *Proceedings of the 2004 SIAM International Conference on Data Mining*. https://www.cs.utexas.edu/~ml/papers/semi-sdm-04.pdf

Basu, S., Davidson, I., & Wagstaff, K. L. (2008). Constrained clustering: Advances in algorithms, theory, and applications. In *Chapman and Hall/CRC data mining and knowledge discovery series* (1st ed.). Chapman & Hall/CRC. https://dl.acm.org/doi/10.5555/1404506

Bensaid, A. M., Hall, L. O., Bezdek, J. C., & Clarke, L. P. (1996). Partially supervised clustering for image segmentation. *Pattern Recognition*, *29*(5), 859–871. doi:10.1016/0031-3203(95)00120-4

Bilenko, M., Basu, S., & Mooney, R. J. (2004). Integrating constraints and metric learning in semi-supervised clustering. *Proceedings of the Twenty-First International Conference on Machine Learning*. doi:10.1145/1015330.1015360

Böhm, C., & Plant, C. (2008). HISSCLU: A hierarchical density-based method for semi-supervised clustering. *Proceedings of the 11th International Conference on Extending Database Technology: Advances in Database Technology*. https://dl.acm.org/doi/pdf/10.1145/1353343.1353398

Chen, Y., Hu, X., Fan, W., Shen, L., Zhang, Z., Liu, X., & Li, H. (2020). Fast density peak clustering for large scale data based on kNN. *Knowledge-Based Systems*, *187*, 104824. doi:10.1016/j.knosys.2019.06.032

Chen, Y., Zhou, S., Zhang, X., Li, D., & Fu, C. (2022). Improved fuzzy c-means clustering by varying the fuzziness parameter. *Pattern Recognition Letters*, *157*, 60–66. doi:10.1016/j.patrec.2022.03.017 PMID:34975183

Cohn, D., Caruana, R., & McCallum, A. (2003). *Semi-supervised clustering with user feedback*. Cornell University, Technical Report TR2003-1892. https://doi.org/10.1201/9781584889977.ch2

Craenendonck, T. V., & Blockeel, H. (2017). Constraint-based clustering selection. *Machine Learning*, *106*(9–10), 1497–1521. https://lin. doi:10.1007/s10994-017-5643-7

Craenendonck, T. V., Meert, W., Dumancic, S., & Blockeel, H. (2018). COBRASTS: A new approach to semi supervised clustering of time series. In Lecture notes in computer science: Vol. 179–193. *International Conference on Discovery Science*. Springer. doi:10.1007/978-3-030-01771-2_12

Dao, T. B. H., Duong, K. C., & Vrain, C. (2017). Constrained clustering by constraint programming. *Artificial Intelligence*, *244*, 70–94. doi:10.1016/j.artint.2015.05.006

Davidson, I., & Ravi, S. S. (2005). Clustering with constraints: Feasibility issues and the k-means algorithm. *Proceedings of the 2005 SIAM International Conference on Data Mining*. doi:10.1137/1.9781611972757.13

Davidson, I., & Ravi, S. S. (2009). Using instance-level constraints in agglomerative hierarchical clustering: Theoretical and empirical results. *Data Mining and Knowledge Discovery*, *18*(2), 257–282. https://lin. doi:10.1007/s10618-008-0103-4

Davidson, I., Ravi, S. S., & Shamis, L. (2010, April 29–May 1). A SAT-based framework for efficient constrained clustering. *Proceedings of the SIAM International Conference on Data Mining*. doi:10.1137/1.9781611972801.9

Ding, L., Xu, W., & Chen, Y. (2020a). Improved density peaks clustering based on natural neighbor expanded group. *Complexity*, *2020*, 1–11. doi:10.1155/2020/8864239

Ding, L., Xu, W., & Chen, Y. (2020b). Density peaks clustering by zero-pointed samples of regional group borders. *Computational Intelligence and Neuroscience*, *2*, 1–15. doi:10.1155/2020/8891778 PMID:32733548

Ertoez, L., Steinbach, M., & Kumar, V. (2003). Finding clusters of different sizes, shapes, and densities in Noisy, high dimensional data. *Proceedings of the 2003 SIAM International Conference on Data Mining*. https://doi.org/ doi:10.1137/1.9781611972733

Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L. M., Agushaka, J. O., Eke, C. I., & Akinyelu, A. A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering Applications of Artificial Intelligence*, *110*, 104743. doi:10.1016/j.engappai.2022.104743

Gilpin, S., & Davidson, I. (2017). A flexible ILP formulation for hierarchical clustering. *Artificial Intelligence*, *244*, 95–109. doi:10.1016/j.artint.2015.05.009

Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.-H., Zhou, Y., Ramaiah, C., Feng, F., Li, R., Wang, S., Athanasakis, D., Shawe-Taylor, J., Milakov, M., Park, J., & Bengio, Y. et al. (2013). Challenges in representation learning: A report on three machine learning contests. In M. Lee, A. Hirose, Z. G. Hou, & R. M. Kil (Eds.), Lecture notes in computer science: Vol. 8228. *Neural information processing. ICONIP 2013* (pp. 117–124). Springer. doi:10.1007/978-3-642-42051-1_16

Grira, N., Crucianu, M., & Boujemaa, N. (2008). Active semi-supervised fuzzy clustering. *Pattern Recognition*, *41*(5), 1834–1844. doi:10.1016/j.patcog.2007.10.004

Hinton, G. E., Krizhevsky, A., & Wang, S. D. (2011). Transforming auto-encoders. In Lecture notes in computer science: Vol. 6791. *ICANN 2011: Artificial Neural Networks and Machine Learning-ICANN* (pp. 44–51). Springer. doi:10.1007/978-3-642-21735-7_6

Hoi, S. C. H., Liu, W., Lyu, M. R., & Ma, W.-Y. (2006, June 17–22). Learning distance metrics with contextual constraints for image retrieval. *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. doi:10.1109/CVPR.2006.167

Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, *31*(8), 651–666. doi:10.1016/j.patrec.2009.09.011

Jarvis, R. A., & Patrick, E. A. (1973). Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on Computers*, *22*(11), 1025–1034. doi:10.1109/T-C.1973.223640

Jiang, Z., Liu, X., & Sun, M. (2019). A density peak clustering algorithm based on the k-nearest shannon entropy and tissue-like P system. *Mathematical Problems in Engineering*, *1713801*, 1–13. Advance online publication. doi:10.1155/2019/1713801

Jonschkowski, R., Höfer, S., & Brock, O. (2015). *Patterns for learning with side information*. https://doi.org//arXiv:1511.06429<ALIGNMENT.qj></ALIGNMENT>10.48550

Klein, D., Kamvar, S. D., & Manning, C. D. (2002). *From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering* [Conference session]. ICML '02: Proceedings of the Nineteenth International Conference on Machine Learning. https://dl.acm.org/doi/10.5555/645531.655989

Kolias, C., Kambourakis, G., Stavrou, A., & Gritzalis, S. (2015). Intrusion detection in 802.11 networks: Empirical evaluation of threats and a public dataset. *IEEE Communications Surveys and Tutorials*, *18*(1), 184–208. doi:10.1109/COMST.2015.2402161

Krishnaswamy, R., Subramaniam, K., Nandini, V., Vijayalakshmi, K., Kadry, S., & Nam, Y. (2023). Metaheuristic based clustering with deep learning model for big data classification. *Computer Systems Science and Engineering*, *44*(1), 391–406. doi:10.32604/csse.2023.024901

Kulis, B., Basu, S., Dhillon, I. S., & Mooney, R. J. (2009). Semi-supervised graph clustering: A kernel approach. *Machine Learning*, *74*(1), 1–22. doi:10.1007/s10994-008-5084-4

Lange, T., Law, M. H. C., Jain, A. K., & Joachim, M. B. (2005). Learning with constrained and unlabelled data. *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. doi:10.1109/CVPR.2005.210

Lelis, L., & Sander, J. (2009). Semi-supervised density-based clustering. *Proceedings of the Ninth IEEE International Conference on Data Mining*. doi:10.1109/ICDM.2009.143

Liang, Z., & Chen, P. (2021). An automatic clustering algorithm based on the density-peak framework and Chameleon method. *Pattern Recognition Letters*, *150*, 40–48. doi:10.1016/j.patrec.2021.06.017

Lin, J. L. (2019). Accelerating density peak clustering algorithm. *Symmetry*, *11*(7), 859. doi:10.3390/sym11070859

Lin, J. L. (2021). Generalizing local density for density-based clustering. *Symmetry*, *13*(2), 185. doi:10.3390/sym13020185

Lin, J. L., Kuo, J. C., & Chuang, H. W. (2020). Improving density peak clustering by automatic peak selection and single linkage clustering. *Symmetry*, *12*(7), 1168. doi:10.3390/sym12071168

Mai, S. T., Amer-Yahia, S., Douzal, A., Nguyen, K. T., & Chouakria, A.-D. (2018). Scalable active temporal constrained clustering. In Lecture notes in computer science: Vol. 10827. *Database systems for advanced applications* (pp. 566–582). Springer. doi:10.1007/978-3-319-91452-7_37

Mai, S. T., Jacobsen, J., Amer-Yahia, S., Spence, I., Tran, N.-P., Assent, I., & Nguyen, Q. V. H. (2022). Incremental density-based clustering on multicore processors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*(3), 1338–1356. doi:10.1109/TPAMI.2020.3023125 PMID:32915725

Mallapragada, P. K., Jin, R., & Jain, A. K. (2008, December 8–11). Active query selection for semi-supervised clustering. *Proceedings of the 19th International Conference on Pattern Recognition*. doi:10.1109/ICPR.2008.4761792

Maraziotis, I. A. (2012). A semi-supervised fuzzy clustering algorithm applied to gene expression data. *Pattern Recognition*, *45*(1), 637–648. doi:10.1016/j.patcog.2011.05.007

Mavroeidis, D. (2010). Accelerating spectral clustering with partial supervision. *Data Mining and Knowledge Discovery*, *21*(2), 241–258. doi:10.1007/s10618-010-0191-9

Mavroeidis, D., & Bingham, E. (2010). Enhancing the stability and efficiency of spectral ordering with partial supervision and feature selection. *Knowledge and Information Systems*, *23*(2), 243–265. doi:10.1007/s10115-009-0215-1

Mehmood, R., Zhang, G., Bie, R., Dawood, H., & Ahmad, H. (2016). Clustering by fast search and find of density peaks via heat diffusion. *Neurocomputing*, *208*, 210–217. doi:10.1016/j.neucom.2016.01.102

Pelleg, D., & Baras, D. (2007). K-means with large and noisy constraint sets. In European Conference on Machine Learning ECML. In *Lecture Notes in Computer Science* (pp. 674–682). Springer. doi:10.1007/978-3-540-74958-5_67

Qian, L., Plant, C., & Böhm, C. (2021, December 7–10). Density-based clustering for adaptive density variation. *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. doi:10.1109/ICDM51629.2021.00158

Rodriguez, A., & Laio, A. (2014). Clustering by fast search and find of density peaks. *Science*, *344*(6191), 1492–1496. doi:10.1126/science.1242072 PMID:24970081

Ruiz, C., Spiliopoulou, M., & Ruiz, E. M. (2010). Density-based semi-supervised clustering. *Data Mining and Knowledge Discovery*, *21*(3), 345–370. doi:10.1007/s10618-009-0157-y

Saha, J., & Mukherjee, J. (2021). CNAK: Cluster number assisted K-means. *Pattern Recognition*, *110*, 107625. doi:10.1016/j.patcog.2020.107625

Sieranoja, S., & Fränti, P. (2019). Fast and general density peaks clustering. *Pattern Recognition Letters*, *128*, 551–558. doi:10.1016/j.patrec.2019.10.019

Tobin, J., & Zhang, M. (2021, December 7–10). DCF: An efficient and robust density-based clustering method. *Proceedings of the 2021 IEEE International Conference on Data Mining (ICDM)*. doi:10.1109/ICDM51629.2021.00074

Vu, V.-V. (2018). An efficient semi-supervised graph-based clustering. *Intelligent Data Analysis*, *22*(2), 297–307. doi:10.3233/IDA-163296

Vu, V.-V., Do, H. Q., Dang, V.-T., & Do, N. T. (2019). An efficient density-based clustering with side information and active learning: A case study for facial expression task. *Intelligent Data Analysis*, *23*(1), 227–240. doi:10.3233/IDA-173781

Vu, V.-V., & Labroche, N. (2017). Active seed selection for constrained clustering. *Intelligent Data Analysis*, *21*(3), 537–552. doi:10.3233/IDA-150499

Vu, V.-V., Labroche, N., & Bouchon-Meunier, B. (2012). Improving constrained clustering with active query selection. *Pattern Recognition*, *45*(4), 1749–1758. doi:10.1016/j.patcog.2011.10.016

Vu, V.-V., Yoon, B., Do, H.-Q., Nguyen, H.-M., Dao, T.-C., Tran, C.-M., & Tran, D.-V. (2022, February 13–16). An empirical study for density peak clustering. *Proceedings of the 24th International Conference on Advanced Communication Technology (ICACT)*. doi:10.23919/ICACT53585.2022.9728922

Wagstaff, K., Cardie, C., Rogers, S., & Schrödl, S. (2001). Constrained K-means clustering with background knowledge. *Proceedings of the Eighteenth International Conference on Machine Learning*. https://dl.acm.org/doi/10.5555/645530.655669

Wang, X., Qian, B., & Davidson, I. (2014). On constrained spectral clustering and its applications. *Data Mining and Knowledge Discovery*, *28*(1), 1–30. doi:10.1007/s10618-012-0291-9

Wang, Y., Wang, D., Zhang, X., Pang, W., Miao, C., Tan, A.-H., & Zhou, Y. (2020). McDPC: Multi-center density peak clustering. *Neural Computing & Applications*, *128*(17), 551–558. doi:10.1007/s00521-020-04754-5

Xie, J., Gao, H., Xie, W., Liu, X., & Grant, P. W. (2016). Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors. *Information Sciences*, *354*, 19–40. doi:10.1016/j.ins.2016.03.011

Xing, E. P., Ng, A. Y., Jordan, M. I., & Russell, S. J. (2002). Distance metric learning with application to clustering with side-information. *Proceedings of the 15th International Conference on Neural Information Processing Systems (NIPS)*. https://dl.acm.org/doi/10.5555/2968618.2968683

Xu, R., & Wunsch, D. I. I. II. (2005). Survey of clustering algorithms. *IEEE Transactions on Neural Networks*, *16*(3), 654–678. doi:10.1109/TNN.2005.845141 PMID:15940994

Yu, S., Wang, Y., Gu, Y., Dhulipala, L., & Shun, J. (2021). ParChain: A framework for parallel hierarchical agglomerative clustering using nearest-neighbor chain. *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, *15*(2), 285–298. doi:10.14778/3489496.3489509

Zhong, C., Malinen, M. I., Miao, D., & Fränti, P. (2015). A fast minimum spanning tree algorithm based on K-means. *Information Sciences*, *295*, 1–17. doi:10.1016/j.ins.2014.10.012

Zhou, R., Zhang, Y., Feng, S., & Luktarhan, N. (2018). A novel hierarchical clustering algorithm based on density peaks for complex datasets. *Complexity*, *2032461*, 1–8. Advance online publication. doi:10.1155/2018/2032461

Viet-Thang Vu is lecturer and the head of Information System division at Hanoi University of Industry, Vietnam. His research interests include cyber-security, image processing, computer vision, machine learning, data mining.

T. T. Quyen Bui is a senior researcher and a head of Department of Automation Technology, Institute of Information Technology (IOIT), Vietnam Academy of Science and Technology (VAST) since May 2015. Dr. Bui's current research interests include embedded systems, robotics, measurement systems, computer vision, brain-computer interface, IoT.

Tien Loi Nguyen is a lecturer in Information Technology at Hanoi University of Industry, Hanoi, Viet Nam. His main research areas are computer vision and natural language processing.

Doan-Vinh Tran received a PhD degree in Informatics Education from the Russian Academy of Educational Sciences, Russia in 1997. Now, he is a lecturer at the Faculty of Educational Technology, University of Education, Vietnam National University, Hanoi. His research concentrates primarily on Informatics Education, Digital Education, applications VR/AR/MR/XR in Digital Education, Clustering and Image processing.

Hong-Quan Do received two M.S. degrees in Information and Communication Technology from University of Science and Technology of Hanoi, Vietnam and The University of Rennes 1, France in 2015. Currently, he works as a lecturer at FPT University in Hanoi, Vietnam. His research concentrates primarily on Clustering, Semi-supervised Clustering, Image processing and Recommender Systems. In addition to his academic pursuits, he actively participates in E-Government projects and contributes to the development of E-Commerce Recommendation applications.

Viet-Vu Vu received a BS in computer science from Hanoi University of Education in 2000, an MS in computer science from Hanoi University of Technology in 2004, and a PhD in computer science from Paris 6 University in 2011. He is a lecturer at CMC University, Hanoi, Vietnam. His research interests include clustering, active learning, semisupervised clustering, and E-government applications.

Sergey M. Avdoshin is a head of the School of Software Engineering in a Higher School of Economics (HSE). His research interests include Cyber Security, Security IT, Information Security, Computer Security, Applied Cryptography Graphs, Software Engineering, Computer Science.