Automatic Neighborhood Search Clustering Algorithm Based on Feature Weighted Density

Tao Zhang, Yuqing He, Decai, Li, Yuanye Xu

***Abstract*—The failure of traditional clustering methods on high-dimensional data has been a thorny problem. Therefore, we propose a simple but effective mean shift feature weighted deformation method (WDNS) to calculate the density value of high-dimensional data points by learning the weights of the features. The neighborhood search is then carried out using the density center in the decision diagram as the starting point, and the points of the same cluster are merged to finally complete the clustering. The experimental results show that the algorithm has higher clustering accuracy than the six existing clustering algorithms. In addition, it has the outstanding feature of automatic parameter setting, which is not available in its peers. In summary, this work can improve the state-of-the-art of clustering algorithms.**

***Index Terms*—Density clustering, Mean shift, High-dimension, Neighborhood, Coalescing.**

I. Introduction

With the development of big data in the Internet of Things, the rapidly increasing volume of data places greater demands on the speed and accuracy of data processing. Clustering is an unsupervised machine learning algorithm [1] that can mine the hidden patterns in the data itself and has a wide range of applications in e-commerce [2], smart manufacturing, geographic information, biogenetics, etc. Some popular paradigms in clustering include center-based approaches such as K-means [3] and its variants [4], hierarchical clustering[5]–[7], spectral clustering [8] [9], density-based methods [10], convex clustering [11], kernel clustering [12], model-based frequentist approaches [13] and Bayesian methods [14].

The majority of the algorithms previously described taking the number of clusters (k) as input by default. However, 1) k might not be known in advance for data from the actual world. A sizable community of relevant scholars has been drawn to the long-standing open challenge of determining k from the dataset itself. Other than that, 2) we also face the challenge of extracting human-readable and useful information content from a data set with hundreds of dimensions.

Researchers have used the mean shift (MS) paradigm [15] to automatically determine the number of clusters and to learn different aspects of the feature space. Mean shift has been utilized in the past for automated feature space grouping, object tracking, and mode finding. There is a wealth of research on clustering high-dimensional data, including algorithms based on data depth [16], bi-clustering [11], dimensionality reduction [17], and subspace clustering [18]. However, the majority of these techniques need expensive computations. Weighted k-means [19] and Sparse k-means [20] have established themselves as benchmark algorithms for learning efficient feature representations of such high-dimensional data while clustering [21]. These techniques lose their attractiveness, though, to practitioners who may have never dealt with data previously and want to determine the number of clusters in an unsupervised way. They also need k as input. In this paper, our purpose is to develop a weighted fuzzy mean drift-based feature weight estimation, an algorithm that automatically finds valid data features to represent high-dimensional data while preserving its computational cost. To achieve this, we introduce a vector of feature weights [22] to learn the importance of each feature when smoothing the data. The resulting iterations yield elegant and simple algorithms with closed-form updates. The weight update scheme follows the idea that features with higher intra-cluster variance contribute less to discovering the clustering structure of the data. The data density is obtained from the feature weight vector, which in turn performs neighborhood search clustering to obtain the final clusters without the need to set them in advance.

The main contributions of this paper can be summarized as follows:

1) We introduce the weighted fuzzy mean shift (WFMS) formula [23], a simple formula that effectively filters out insignificant features from the data.

2) We propose a neighborhood search clustering method with an adaptive neighborhood. The neighborhood radius of each dataset is only related to the density of data points, which can be automatically calculated, thus avoiding manual settings.

3) Through detailed experimental analysis, we demonstrate the effectiveness of our proposed Weighted Density Neighborhood Search algorithm (WDNS) on simulated and real data against state-of-the-art clustering techniques.

Manuscript received December 31, 2022; revised March 4, 2023; accepted April 6, 2023.

Tao Zhang is with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and with the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China.

Yuqing He and Decai Li are with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and also with the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China.

Yuanye Xu is with the Key Laboratory of Networked Control System, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China, and also with the Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China.

*Correspondence: zhantao1@sia.cn

doi: 10.18178/ijke.2023.9.1.137

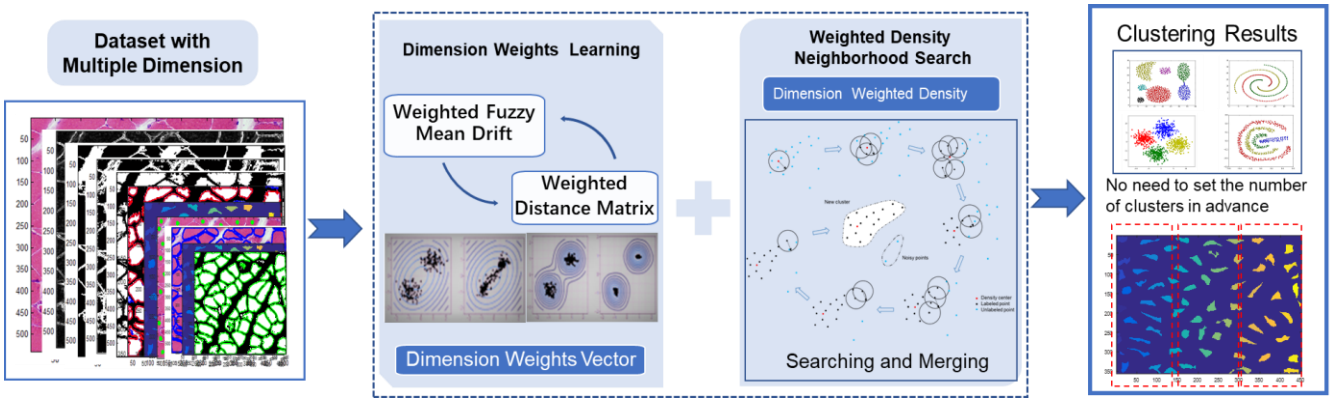


Fig. 1. Schematic of the proposed WDNS framework.

II. Method

In this section, we give the framework module for the Weighted Density Neighborhood Search algorithm: a high-dimensional mean drift algorithm is introduced into the calculation of the density to obtain the dimensional weight density; clustering is performed by searching and merging within the neighborhood of the data to automatically obtain the corresponding number of clusters without prior setting. We give some important concepts, especially the definition of density. In it, outlier detection, backbone identification, and density definition depend on the notion of *Eps* (the cutoff distance) and γ (the density threshold). Throughout this paper, all definitions are based on dataset $\boldsymbol{D}=\{p_1, p_2, \ldots, p_n\}$, containing d dimensions, where p_i, $i \in \{1, 2, \ldots, n\}$, is a point in D. $N_p(\mu, \Sigma)$ denotes the variate normal distribution with mean μ and dispersion matrix Σ.

A. Preliminaries

Mean Shift (MS) is a non-parametric method based on density gradient ascent that finds the target position and achieves target tracking by iterative operations. Also, it is possible to find all dense regions of data points based on a sliding window algorithm. That is, it aims to locate the centroids of each cluster, which is done by updating the candidate points of the centroids to the mean value of the points within the sliding window. These candidate windows are then filtered in the post-processing stage to eliminate near-duplicates, resulting in a final set of centroids and their corresponding groups.

$$M(p) = \frac{1}{k} \sum_{p_i \in S_h} (p_i - p) \tag{1}$$

$M(p)$: Offset of the mean iteration.

S_h: A region of high-dimensional spheres of radius h with p as the center.

k: number of points contained in the range of S_h.

p_i: points contained in the range of S_h.

Centre update: (moves the center points in the direction of the vector of offset means)

$$p^{t+1} = M^t + p^t \tag{2}$$

In 2014, Alex and Alessandro [24] proposed a new clustering algorithm. All centers of each cluster are selected from the samples in the dataset. As with the mean-shift method, the cluster centers are defined as local maxima of the data point density. However, unlike the mean-drift method, the procedure does not require embedding the data in a vector space and explicitly maximizing the density field of each data point.

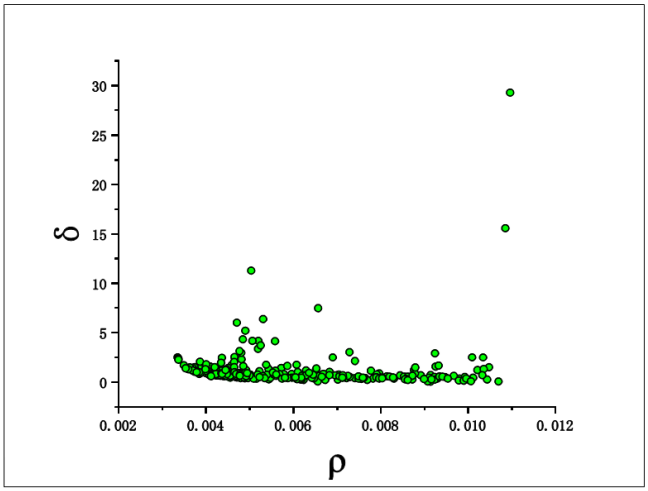


Fig. 2. A decision graph of dataset "path-based".

The algorithm has assumptions that neighbors surround cluster centers with lower local density, and they are at a relatively large distance from any points with a higher local density. For each data point i, we compute two quantities: its local density ρ_i and its distance δ_i from points of higher density. Both quantities depend only on the distances between data points, which are assumed to satisfy the triangular inequality.

For point i in dataset D, if ρ_i is not the largest, then δ_i is the minimum distance between i and other points with a higher density than i.

$$\delta_i = Min\{d_{ij}\} \qquad j : \rho_i < \rho_j \tag{3}$$

If ρ_i is the largest, δ_i is the maximum distance between i and other points in D.

Let us use the megalopolis as an analogy to explain the distance δ as Fig. 3. Megalopolis is composed of several large or small cities, and a cluster consists of many samples. The central cities of megalopolis are like the density centers of clusters, and city size can be likened to ρ, the density of samples.

Fig. 3. Urban distribution in some megalopolis of China and USA

In order to become the central metropolis of a megalopolis, the city must be large enough and far enough away from other larger cities. Shanghai and New York are the central cities of their megalopolis, and there are also many big cities around them, such as Hangzhou and Philadelphia. Because they are adjacent to larger cities, Hangzhou and Philadelphia cannot become the central cities of the megalopolis. However, cities like Wuhan and Jacksonville, comparable in size but far removed from Shanghai and New York, have built megalopolis around themselves. Therefore, the distance from other larger cities becomes the key to becoming a metropolis. For samples in clustering, megalopolis distance δ is the key to being a cluster center as well.

Dataset "path-bases" is embedded in a two-dimensional space where the x-coordinate represents ρ_i and the y-coordinate represents δ_i, as shown in Fig. 2. We call this representation a Decision Graph. It can be utilized for selecting centers of clusters with the advantages of simplicity, intuitiveness, and accuracy.

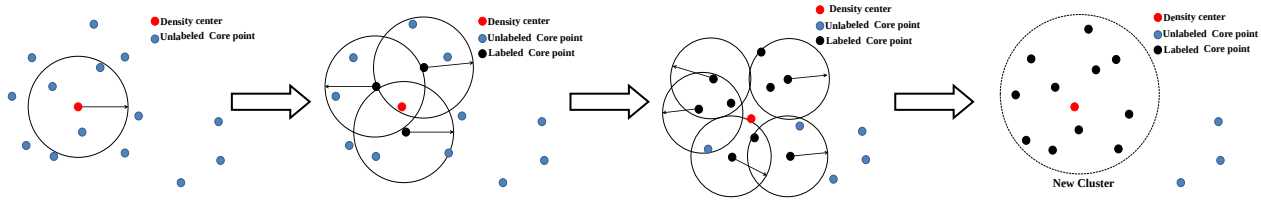
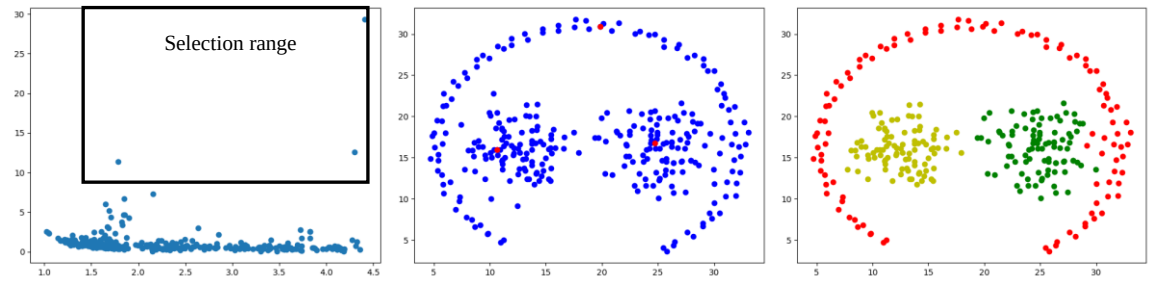
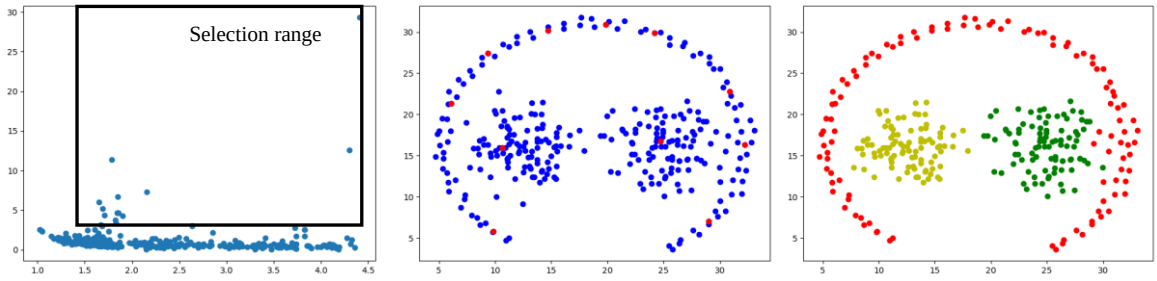


Fig. 4. Processes of neighborhood searching.



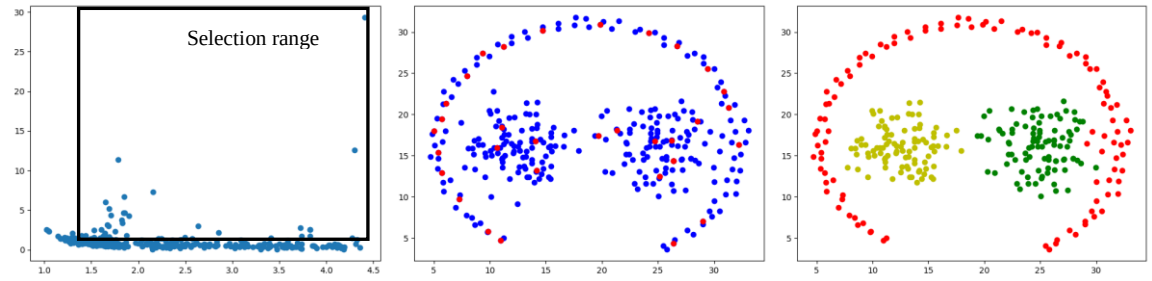
3 selected density centers in a decision graph Density centers Clustering result

(a) The process of clustering when we select just 3 density centers in the decision graph, α= 1.5, β= 9, and γ= 1.32.



11 selected density centers in a decision graph Density centers Clustering result

(b) The process of clustering when we select 11 density centers in the decision graph, α= 1.5, β= 2.6, and γ= 1.32.



30+ selected density centers in a decision graph Density centers Clustering result

(c) The process of clustering when we select 30+ density centers in the decision graph, α= 1.5, β= 1.5, and γ= 1.32.

Fig. 5. The robustness of the algorithm in choosing density centers. In dataset Path-based1, we use three different methods of (a), (b), and (c) to select density centers on the decision graph. For each operation, such as the operation of selecting 3 density centers in (a), density centers, which are selected in the box, are shown in the second graph with red color, while the others are blue, and the clustering result is shown in the third graph.

B. Formulation

We will use a similar update rule as in MS (equation (1-2)). However, instead of the usual Euclidean distance, we will use the weighted distance $\| \ \|_w$. The update rule for the data points is given by,

$$p_i^{t+1} = \frac{\sum_{j=1}^{n} (\left\| p_i^t - p_j^t \right\|_{\omega^t}) p_j^t}{\sum_{j=1}^{n} (\left\| p_i^t - p_j^t \right\|_{\omega^t})} \tag{4}$$

The feature weights are updated as follows:

$$\omega_l^t = \frac{1 - \exp\left\{ -\frac{1}{n} \sum_{i=1}^{n} \left(x_{il} - y_{il}^t \right)^2 \right\}}{\sum_{l=1}^{d} \left(1 - \exp\left\{ -\frac{1}{n} \sum_{i=1}^{n} \left(x_{il} - y_{il}^t \right)^2 \right\} \right)} \tag{5}$$

where:

$$\|x - y\|_\omega = \sqrt{\sum_{l=1}^{d} \omega_l (x_l - y_l)^2} \tag{6}$$

A non-parametric density for point *i* is calculated as:

$$\rho_i = \sum_{j \neq i}^{n} \exp\left\{ -d_{ij} / \bar{d} \right\} \tag{7}$$

where:

$$\bar{d} = \frac{1}{n} \sum_{\substack{i=1 \\ i \neq j}}^{n} \min(d_{ij}) \tag{8}$$

In this paper, the end condition of the iteration is $\bar{d}$ =0. The density value calculated at $\bar{d}$ is the weighted density of the data points in the multidimensional data set, neighborhood searching of density centers is performed in the next step.

Density center: points in *D* satisfying the following conditions belong to a set of density centers D^c:

$$D^c = \{ i \mid \delta_i \geq \alpha, \rho_i \geq \beta, \text{and } i \in D \} \tag{9}$$

One of the crucial elements in our clustering procedure is the selection of density centers, where α and β are two thresholds, which are parameters determined by a decision graph. If they were poorly chosen, clustering accuracy would suffer, maybe even failing altogether. As density centers, we here choose the places with big δ and ρ. This point is more away from other dense spots the larger δ, making it more likely to develop as a clustering center. ρ of a density center must also be sufficiently large; if not, it is most likely to be a noise point. Contrary to DPC, the centers we choose are not clustering centers. Therefore, we don't require exact density centers. Instead, we only need to specify their range approximately, and the inside points can be later elected as density centers.

Neighborhood search is a method for looking for points within a hypersphere whose radius is determined by the amount of data available, and where all of the points are grouped into the same cluster. A clustering technique uses it as a recursive procedure.

Another critical phase of a clustering process is coalescing via neighborhood search, as seen in Fig. 4. The chosen density centers are then organized in terms of density. Second, the neighborhood search begins at the location with the highest density, and all of its neighbors are grouped as one cluster. Third, until no other neighbors are discovered, this cluster is expanded by including the vicinity of the combined points. We, therefore, have the first cluster. Once all density centers have been identified, the next density center is chosen to coalesce the remaining rest points for another cluster. A point is considered noise if it is not merged into any clusters.

Merging points by searching the neighborhood of density centers, which brings great robustness to it. We take the dataset "Path-based1", a synthetic data set, as an example. We use three different methods of (a), (b) and (c) to select density centers on a decision graph (Fig. 5). For each operation, such as the operation of selecting 3 density centers in Fig. 5 (a), the density centers, as selected in the box, are shown in the second graph in red while the others are in blue, and the clustering result is shown in the third graph. We can see that the three methods lead to the same clustering result. Thus, algorithm has great flexibility in selecting density centers. In order to ensure the accuracy of clustering, it is appropriate to select more density centers. However, the operating speed is slightly reduced.

C. Convergence Analysis

Our proposed weight update scheme can be considered as a sum of density. Thus, the objective function is given by

$$\begin{aligned} \sum_{i=1}^{n} \rho_i^t &= \sum_{i=1}^{n} \sum_{\substack{j \neq i \\ i=1}}^{n} \exp\left\{ -d_{ij}^t / \bar{d}^t \right\} \\ &= \sum_{i=1}^{n} \sum_{\substack{j \neq i \\ i=1}}^{n} \exp\left\{ -d_{ij}^t / (\frac{1}{n} \sum_{\substack{j \neq i \\ i=1}}^{n} \min d_{ij}^t) \right\} \end{aligned} \tag{10}$$

It is strictly monotonically increasing, it can be proved that the iteration process converges, and it can be deduced that:

$$\left| \sum_{i=1}^{n} d_{ij}^{t+1} - \sum_{i=1}^{n} d_{ij}^t \right| < \delta \tag{11}$$

Assuming that:

$$f(p^i) = \sum_{i=1}^{n} \rho_i^t \tag{12}$$

Using a second-order Taylor Expansion at $\bar{p}$, we get

$$\begin{aligned} f(p) = & f(\bar{p}) + \nabla f(\bar{p})^T (p - \bar{p}) \\ & + \frac{1}{2} (p - \bar{p})^T \nabla^2 f(\hat{p}) (p - \bar{p}) \end{aligned} \tag{13}$$

$$\hat{p} = p + \eta(p - \bar{p}), 0 < \eta < 1 \tag{14}$$

If $p = p^t$, we can get

$$f(p^t) - f(\bar{p}) = \frac{1}{2} (p^t - \bar{p})^T \nabla^2 f(\hat{p}) (p^t - \bar{p}) \tag{15}$$

Because $f(\bar{p}) > f(p^t)$, so

$$(p^t - \bar{p})^T \nabla^2 f(\hat{p}) (p^t - \bar{p}) < 0 \tag{16}$$

There are

$$\begin{aligned} & \lim_{\eta \to 1} (p^t - \bar{p})^T \nabla^2 f(\hat{p}) (p^t - \bar{p}) \\ & = (p^t - \bar{p})^T \nabla^2 f(p^t) (p^t - \bar{p}) < 0 \end{aligned} \tag{17}$$

So, $\nabla^2 f(p^t)$ is a positive definite, which is contrary to

being a positive semidefinite matrix. Then, $f(p^t)$ $t = 1,2 \ldots$ and strictly monotonically increasing.

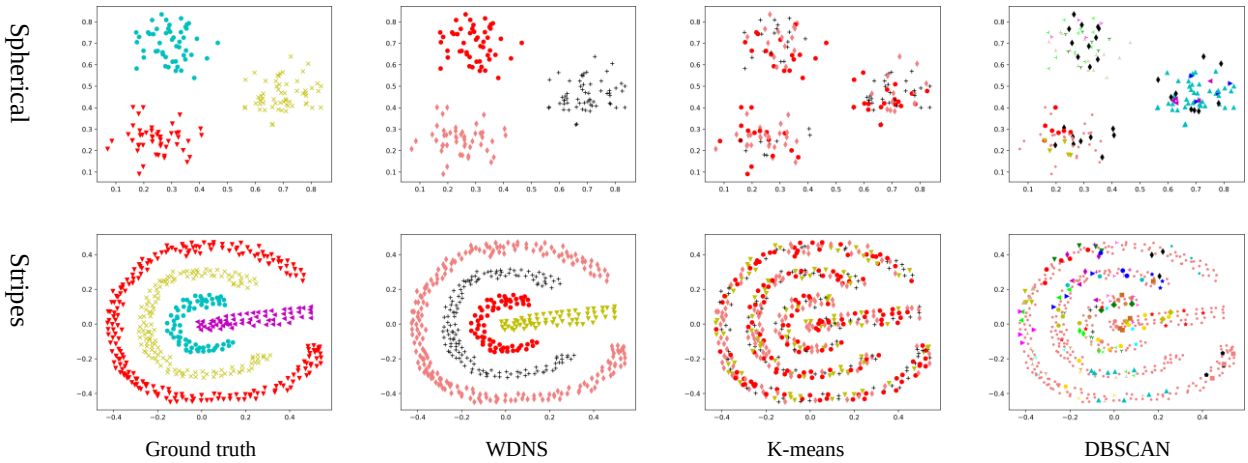


Fig. 6. Results of neighborhood searching.

D. Time Complexity Analysis

The time complexity of WNDS mainly consists of the dimensional weight density calculation part and neighborhood search clustering part. The time complexity of the first part is $O(2n^2ds)$, where s denotes the number of iterations in this part. The neighborhood search clustering part includes selecting density centers decided by density calculation, whose time complexity is $O(dn^2)$, and neighborhood searching, whose complexity is $O(dnm)$, where m is the number of points as i's neighbors. Therefore, the time complexity of WNDS is $O(2n^2ds+dn^2)$ since $m \ll n$.

E. Weighted Density Neighborhood Search Algorithm

Algorithm WDNS: (D, α, β, γ)
Input: A dataset D containing n samples.
α, β: These two thresholds are used to select δ and ρ.
γ: The threshold of selecting core points.
Initialization: $\omega^0 = [1,1,\ldots,1]$
Output: Density-based clusters.

repeat
  for i j **in** D
    $d_{ij} = \|p_i - p_j\|_{\omega^0}$ (refer to Equation 6)
  end
  Calculate $\bar{d}$ (refer to Equation 8)
  If $\bar{d} \neq 0$
    update p_i^t (refer to Equation 4)
    update ω_i^t (refer to Equation 5)
    $D = D^t$
  else
    break
for i **in** D
  Calculate ρ_i (refer to Equation 7)
Create a density dictionary with i and its density
for d_{ij} **in** Euclidean distance matrix
  find max d_{ij}
end
for I, j **in** Density dictionary
  if $\rho_i < \rho_j$:
    δ_i = min(d_{ij})
  else
    δ_i = max (d_{ij})
end
for i **in** D
  if $\delta_i > \alpha$ and $\rho_i > \beta$
    Density centers← i
end
for i **in** D
  if $\rho_i > \gamma$
    Core points← i
  else
    Noise points← i
end
for i **in** Core points
  for j **in** D
    if $d_{ij} < \bar{d}$
      Set(neighborhood of i)← j
  end
end
Density centers ranged by ρ
for i **in** Density centers
  list= neighborhood of i
  for j **in** list
    if j **in** core points
      j **goto** cluster i
  end
  list=*list1*
  if *list* is disposed all
    break
end

III. Experimental Results

We compared WDNS with classical and excellent clustering algorithms and verified the accuracy and adaptability of WDNS through experiments. Two synthetic datasets and eight UCI real datasets were selected. The synthetic and UCI datasets are numerical datasets consisting of multidimensional attributes. All experiments were performed on a laptop with 64-bit Windows, core i5 CPU, and 16 GB RAM running Python 3.7.

A. Artificial Dataset Analysis

In this part, we generated two artificial datasets, the first one with 300 points and 22 dimensions, called Spherical, which consists of three clusters of 100 points each. spherical cluster structure is fully contained in the first two features, and the remaining 20 features are generated independently from the standard normal distribution without cluster information. The second one has 500 points and 12 dimensions, called Stripes, which consists of five clusters, cluster structure is fully contained in the first two features too, rest are generated independently as well. Figs. 4 show the benchmarked results of the three algorithms on the artificial datasets respectively. The first subfigure in each row is a dataset label, and the dots with different colors mean that they belong to different clusters. The second to fourth column figures are the results of WDNS, K-means [4], and DBSCAN [10], respectively.

B. Applications for the Face Clustering

The ORL Database has 40 distinct individuals, each

with 10 different images. For these individuals, the images are taken at different time and with different lighting, facial expressions (open/closed eyes, smiling/not smiling), and

Fig. 7. ORL face dataset

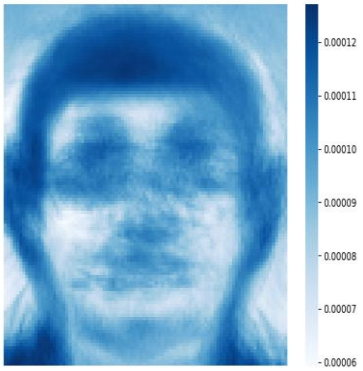


Fig. 8. Weight matrix

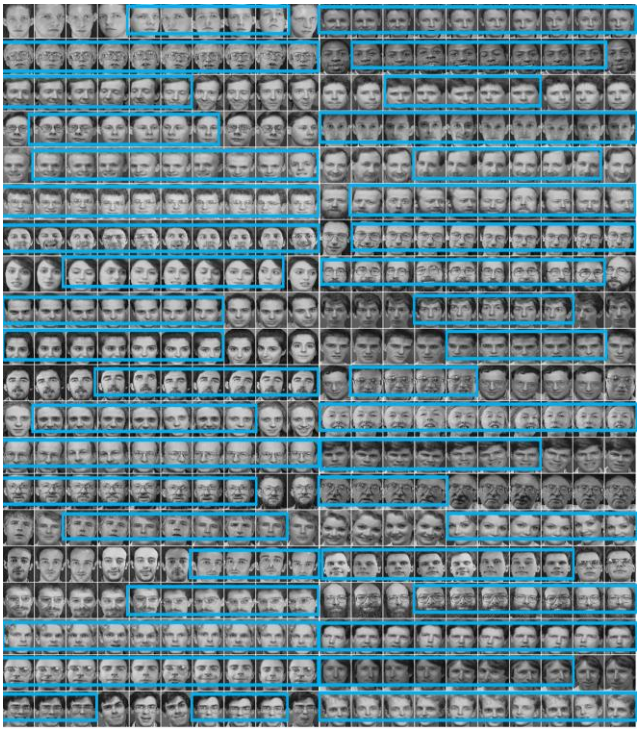


Fig. 9. ORL dataset cluster result.

facial details (glasses/no glasses). All the images are taken against a dark homogeneous background with the individuals in an upright, frontal position (with tolerance for some side movement). Each image has 92×112 pixels, and each pixel has a gray value of 0-256. With each pixel as the feature, each image has 10304 dimensions as in Fig. 7, and the input of the algorithm is 400 vectors with 10304-dimensions.

Through iteration, the weight matrix obtained is shown in Fig. 8, with 10304 attributes, and the weight of each attribute is between 0.0006 and 0.0013. The darker the color in the figure, the higher the weight of the attribute. It can be seen that the outer circle contour and the weight of the facial features in the image are relatively large, indicating that these attributes carry more information. In the subsequent density calculation and neighborhood search, the noisy attributes and irrelevant attributes can be eliminated by filtering the weight matrix, which improves the robustness of the algorithm and makes the clustering result more accurate.

The clustering results are shown in Fig. 9. Faces in the same blue box belong to the same cluster, and the rest images unmarked do not belong to any cluster.

C. Comparison against Base Clusterings

After obtaining its remarkable clustering effect on synthetic datasets, we compare it with other state-of-the-art traditional clustering algorithms, including DBSCAN, K-means, DPC, FCAN, CNN, FCM-VMF[25], and niMM[26]. Table I lists the description of ten multi-dimensional datasets, whose dimensions range from 36 to 3231961. Each dataset contains a different number of samples and classes, with strong typicality and universality.

TABEL I. UCI DATASETS

Datasets	features	classes	samples
Satimage	36	6	6435
Arrhythmia	249	16	452
M-feat	240	10	2000
Hill_Valley	100	2	606
Urban land cover	506	9	148
Parkinson's Disease	754	2	756
Asian Religious and Biblical Texts	8265	8	590
Arcene	10000	2	900
Dorothea	100000	2	1950
URL Reputation	3231961	2	2396130

Satimage dataset consists of multi-spectral values of pixels in 3x3 neighborhoods for each satellite image. M-feat dataset is a collection of handwritten numbers containing "0-9". In Hill-Valley dataset, each record represents 100 points on a two-dimensional graph. The Urban Land Cover dataset contains training and testing data to classify a high-resolution aerial image into nine urban land cover types. The Parkinson's Disease Classification dataset has 754 attributes and 756 instances. Most of the sacred texts in Asian Religious and Biblical Texts were collected from Project Gutenberg. Three mass spectrometry datasets constitute Arcene dataset to obtain sufficient training and test data. The Dorothea dataset is a collection of drug discovery data. The URL Reputation dataset is a 120-day anonymized subset of the URL data.

From Table II and Fig. 10, all other algorithms have some defects except WDNS in cluster accuracy. WDNS has obvious advantages in the NMI index, which exceeds its comparison algorithm to a large extent. The calculation of ARI and RI are similar, so there is a certain convergence in the clustering accuracy of these datasets. Note that for Arrhythmia, K-means and DPC perform poorly on ARI and RI metrics because this dataset is nonlinear. The Asian Religious and Biblical Texts are quite difficult, and the performance of FCM-VMF, CNN, DPC, and niMM are all disappointing. Arcene is just the opposite, these algorithms all perform about the same, probably related to the fact that the dataset has fewer categories. For the rest of the datasets, each algorithm has its strengths and weaknesses. So, a

statistical test for the comparison in Table II is carried out utilizing nonparametric testing for multiple comparisons to confirm the advantage of WDNS.

The Friedman test, a nonparametric alternative to repeated-measures ANOVA, is frequently used to assess the overall effectiveness of *k* algorithms on *N* datasets.

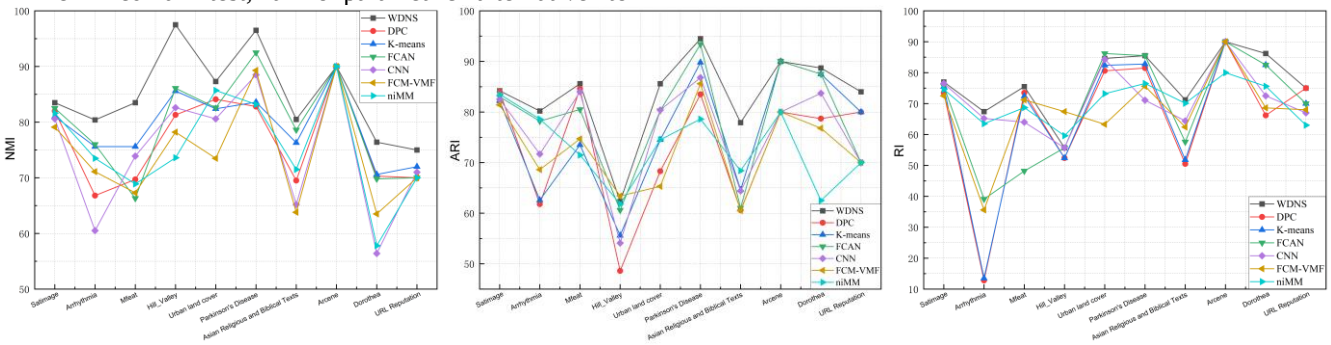


Fig. 10. Clustering accuracy of WDNS compared with other algorithms on UCI datasets.

TABEL II. THE CLUSTERING PERFORMANCE FOR UCI DATASETS (THE BEST SCORES IN EACH ROW ARE HIGHLIGHTED IN BOLD)

Datasets	Methods	WDNS	DPC	K-means	FCAN	CNN	FCM-VMF	niMM
Satimage	NMI	**83.5**	82.0	81.0	82.5	80.6	79.1	81.6
	ARI	**84.2**	84.0	82.2	83	82.5	81.4	83.5
	RI	**77**	73.5	75.5	75.7	76.5	72.5	74.6
Arrhythmia	NMI	**80.4**	66.8	75.6	76.0	60.5	71.1	73.5
	ARI	**80.2**	61.8	62.6	78.2	71.7	68.6	78.6
	RI	**67.4**	12.8	13.5	39.1	65.2	35.6	63.5
M-feat	NMI	**83.5**	69.7	75.6	66.3	73.9	67.3	68.9
	ARI	**85.6**	84.6	73.5	80.5	84	74.7	71.5
	RI	**75.5**	73.5	72.4	48.2	64	71.2	68.7
Hill_Valley	NMI	**97.5**	81.3	85.6	86.1	82.6	78.2	73.6
	ARI	**62.3**	48.6	55.6	60.6	54.1	63.4	61.8
	RI	**55.7**	52.4	52.4	55.7	55.7	67.4	59.6
Urban land cover	NMI	**87.3**	84.1	82.4	82.6	80.6	73.5	85.7
	ARI	**85.6**	68.3	74.5	80.3	80.4	65.3	74.5
	RI	**84.6**	80.6	82.4	86.2	84.3	63.3	73.2
Parkinson's Disease	NMI	**96.5**	82.8	83.6	92.5	88.5	89.3	83.2
	ARI	**94.5**	83.5	89.8	93.4	86.8	85.6	78.6
	RI	**85.5**	81.5	82.8	85.5	71.1	75.6	76.5
Asian Religious and Biblical Texts	NMI	**80.5**	69.5	76.3	78.6	65.2	63.8	71.5
	ARI	**77.9**	60.6	64.6	61	64.4	60.5	68.4
	RI	**71.2**	50.5	51.8	57.6	64.4	62.4	70.1
Arcene	NMI	90	90	90	90	90	90	90
	ARI	90	80	90	90	80	80	80
	RI	90	90	90	90	90	90	80
Dorothea	NMI	**76.4**	70.3	70.6	69.8	56.4	63.5	57.8
	ARI	**88.7**	78.7	87.5	87.5	83.7	76.8	62.5
	RI	**86.2**	66.2	82.5	82.5	72.5	68.6	75.6
URL Reputation	NMI	**75**	70	72	70	71	70	70
	ARI	84	80	80	70	70	70	70
	RI	**75**	75	70	70	67	68	63

TABEL III. RANKS AND P-VALUES OF THE COMPARED ALGORITHMS FOR THE BENCHMARK DATASETS

	WDNS	DPC	K-means	FCAN	CNN	FCM-VMF	niMM
Satimage	1	5	6	2	4	7	3
Arrhythmia	1	7	6	4	3	5	2
M-feat	1	2	4	7	3	5	6
Hill_Valley	1	7	5	3	6	2	4
Urban land cover	1	6	4	2	3	7	5
Parkinson's Disease	1	5	3	2	6	4	7
Asian Religious and Biblical Texts	1	7	5	3	4	6	2
Arcene	1	4	1	1	4	4	7
Dorothea	1	4	2	3	5	6	7
URL Reputation	1	2	3	4	5	5	7
Average rank (R)	1	4.9	3.9	3.1	4.3	5.1	5
z		2.07	1.64	1.31	1.81	2.15	2.11
P-values		9.58367E-07	9.61927E-05	0.001935206	1.70798E-05	3.39653E-07	5.73303E-07
Critical values		0.017	0.010	0.008	0.013	0.050	0.025

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \quad (18)$$

R_j is the average rank of the algorithm, $R_j = \frac{1}{N}\sum_i r_i^j$. We can find $F_{(0.05)}[7,10]$ =3.135 in Friedman test critical value table for a smaller number of algorithms and data sets like our experiment. Based on each dataset's average ranks for three performance metrics (Table III), we get χ_F^2=15.7 >3.135. Therefore, the null hypothesis of the test can be rejected at the significance level of 0.05. It is clear that the clustering capabilities of the seven techniques greatly.

$$z = (R_i - R_j) \Big/ \sqrt{\frac{k(k+1)}{6N}} \tag{19}$$

where the Average rank (R) is the rank of the compared peer, we use WDNS as the control algorithm, P-value could be computed through normal approximations. Let p_1, p_2, …, p_{k-1} be the P-value sorted in ascending order. where p_{k-1} is the largest P-value. Each p_i corresponds to hypothesis H_i. Starting from $i=1$, hypotheses H_i is rejected when $p_i < \alpha/(k-i)$. Using α=0.05, Table III shows the sorted P-values and their critical values. As all P-values are less than their corresponding critical values, all hypotheses are rejected, indicating that WDNS outperformed other algorithms with statistical significance.

IV. Discussions

The WDNS algorithm proposed in this paper has two contributions. Firstly, we introduce a weighted fuzzy mean shift (WFMS) formulation which can effectively filter out insignificant features from the data and thus obtain an accurate weighted density calculation method. Secondly, a clustering method based on neighborhood search is proposed, in which the selected density centers are used as the starting point for neighborhood search, and the core points are searched in their neighborhood until there are no core points in the neighborhood. A detailed experimental analysis of simulated and real data shows that WDNS is particularly useful for high-dimensional data with multiple clusters.

Funding

This work was supported by the National Natural Science Foundation of China (61991413, 91948303); Joint Fund of Natural Science Foundation (U22B2041); Innovation Research Group Science Fund of National Natural Science Foundation of China (61821005); Liaoning province central government guides local science and technology development special project (2022JH6/100100009).

References

[1] O. M. Jafar and R. Sivakumar, "Ant-based clustering algorithms: A brief survey," *International Journal of Computer Theory and Engineering*, vol. 2, no. 5, p. 787, 2010.

[2] N. Laloo and M. S. Sunhaloo, "Adapting distance based clustering concept to a heterogeneous network," *International Journal of Computer Theory and Engineering*, vol. 7, no. 3, p. 214, 2015.

[3] G. Armano and M. R. Farmani, "Clustering analysis with combination of artificial bee colony algorithm and k-means technique," *International Journal of Computer Theory and Engineering*, pp. 141–145, 2014.

[4] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[5] G. Carlsson and F. Mémoli, "Characterization, stability and convergence of hierarchical clustering algorithms," 2010.

[6] K. Srivastava, R. Shah, D. Valia, and H. Swaminarayan, "Data mining using hierarchical agglomerative clustering algorithm in distributed cloud computing environment," *International Journal of Computer Theory and Engineering*, vol. 5, no. 3, p. 520, 2013.

[7] Y. C. Hsu, Y. C. Li, and Y. H. Lin, "Toward understanding the user behavior in sports university library using hierarchical clustering," *IJCTE*, Department of Sports Information and Communication, National Taiwan University of Sport, Taichung, Taiwan, vol. 12, no. 4, pp. 97–101, 2020, doi: 10.7763/IJCTE.2020.V12.1271.

[8] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems*, vol. 14, 2001.

[9] S. Hess, W. Duivesteijn, P. Honysz, and K. Morik, "The SpectACl of nonconvex clustering: A spectral approach to density-based clustering," in *Proc. the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3788–3795.

[10] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise.," in *kdd*, 1996, vol. 96, no. 34, pp. 226–231.

[11] E. C. Chi, G. I. Allen, and R. G. Baraniuk, "Convex biclustering," *Biometrics*, vol. 73, no. 1, pp. 10–19, 2017.

[12] I. S. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *Proc. the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 551–556.

[13] P. D. McNicholas, "Model-based clustering," *Journal of Classification*, vol. 33, no. 3, pp. 331–373, 2016.

[14] H. P. Kriegel, P. Kröger, and A. Zimek, "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering," *Acm Transactions on Knowledge Discovery from Data (tkdd)*, vol. 3, no. 1, pp. 1–58, 2009.

[15] M. Su and Y. Shang, "Solving fixed-point problems with inequality and equality constraints via a non-interior point homotopy path-following method," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–9, 2017, doi: 10.1155/2017/3456834.

[16] S. Sarkar and A. K. Ghosh, "On perfect clustering of high dimension, low sample size data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 9, pp. 2257–2272, 2019.

[17] J. Jin and W. Wang, "Influential features PCA for high dimensional clustering," *The Annals of Statistics*, vol. 44, no. 6, pp. 2323–2359, 2016.

[18] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2765–2781, 2013.

[19] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in k-means type clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 657–668, 2005.

[20] D. M. Witten and R. Tibshirani, "A framework for feature selection in clustering," *Journal of the American Statistical Association*, vol. 105, no. 490, pp. 713–726, 2010.

[21] D. CAI and B. FAN. "Spatial feature selection for underwater object detection," *Information and Control*, vol. 51, no. 2, pp. 214-222, 2022.

[22] S. Chakraborty, D. Paul, and S. Das, "Automated clustering of high-dimensional data with a feature weighted mean shift algorithm," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, Art. no. 8, May 2021.

[23] N. K. Visalakshi, K. Thangavel, and R. Parvathi, "An intuitionistic fuzzy approach to distributed fuzzy clustering," *International Journal of Computer Theory and Engineering*, vol. 2, no. 2, p. 295, 2010.

[24] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[25] H. Zhang, H. Li, N. Chen, S. Chen, and J. Liu, "Novel fuzzy clustering algorithm with variable multi-pixel fitting spatial information for image segmentation," *Pattern Recognition*, vol. 121, p. 108201, Jan. 2022, doi: 10.1016/j.patcog.2021.108201.

[26] X. Fan, R. Y. D. Xu, L. Cao, and Y. Song, "Learning nonparametric relational models by conjugately incorporating node information in a network," *IEEE Transactions on Cybernetics*, vol. 47, no. 3, pp. 589–599, Mar. 2017, doi: 10.1109/TCYB.2016.2521376.