An Improved Density Peak Clustering Algorithm

Jian Hou[1(✉)] and Xu E[2]

[1] College of Engineering, Bohai University, Jinzhou 121013, China
dr.houjian@gmail.com
[2] College of Information Science, Bohai University, Jinzhou 121013, China

Abstract. Density based clustering is an important clustering approach due to its ability to generate clusters of arbitrary shapes. Among density based clustering algorithms, the density peak (DP) based algorithm is shown to a potential one with some attractive properties. The DP algorithm calculates the local density of each data, and then the distance of each data to its nearest neighbor with higher density. Based on these two measurements, the cluster centers can be isolated from the non-center data. As a result, the cluster centers can be identified relatively easily and the non-center data can be grouped into clusters efficiently. In this paper we study the influence of density kernels on the clustering results and present a new kernel. We also present a new cluster center selection criterion based on distance normalization. Our new algorithm is shown to be effective in experiments on ten datasets.

Keywords: Density peak · Cluster center · Distance normalization

1 Introduction

Clustering refers to the task of grouping data into clusters based on their similarity, so that the similar data are in the same cluster and dissimilar ones are in different clusters. While there are already a large amount of clustering algorithms of different types, in this paper our work is based on the density peak (DP) based clustering algorithm presented in [26].

The DP algorithm is a density based clustering approach proposed recently and is shown to be attractive in clustering tasks. However, the original DP algorithm cannot be regarded as a reliable clustering approach. First, the density kernel and involved parameters have a significant influence on the density calculation, and then on the final clustering results. In [26] the cutoff kernel and Gaussian kernel are tested, and it is found that both the kernel types and cutoff distance impact on the clustering results. This means that we may need a careful tuning process to obtain satisfactory clustering results. Second, while cluster centers can be isolated from non-center data based on ρ and δ in theory, the identification of cluster centers in practical tasks is much more difficult than it seems. In order to avoid determining thresholds for both ρ and δ, [26] proposes

H. Yin et al. (Eds.): IDEAL 2017, LNCS 10585, pp. 211–221, 2017.
https://doi.org/10.1007/978-3-319-68935-7_24

to use $\gamma = \rho\delta$ as the sole criterion of cluster center selection, and data with large γ have larger probability to be identified as cluster centers. This method is found to be afflicted by three major problems. The first is that by multiplying ρ by δ, some non-center data with very large ρ or very large δ may have a large γ and therefore be recognized as cluster centers by mistake. Even if the non-center data have smaller γ than cluster centers, their difference may not be evident enough and the cluster centers cannot be identified automatically. In this case, we need the number of clusters to select the cluster centers. A third problem is that this criterion fails to take the density difference among clusters into account, and the centers of small-density clusters have a small chance to be identified correctly.

In this paper we improve the original DP algorithm to obtain better clustering results. In order to solve the problems caused by density difference among clusters, [15] proposes to normalize the density ρ based on the density of neighboring data. In this paper we resort to a different approach to solve this problem. By studying the relationship between δ and cluster centers, we propose to use the normalized δ as the criterion of cluster center selection, in contrast to [15] where $\gamma = \rho\delta$ is still the final criterion. In addition, we present a new density kernel in density calculation. We experiment with ten datasets and validate the effective of the proposed approach.

2 Related Works

By grouping a set of data into a number of clusters, the implicit data distribution pattern can be identified and utilized in further processing. Data clustering has wide application in various fields including data mining, pattern recognition, machine learning and image processing. Typical examples include the application in social networks [8,17], music [7], virtual worlds [9] and bioinformatics [23].

A vast amount of clustering algorithms have been proposed and some of them are reviewed briefly here. The k-means algorithm is one of the most simple and most commonly used clustering approach, and a lot of variants [18,19,21,22] have been proposed to improve on the original one. The DBSCAN [5] algorithm is one of the most popular density based clustering algorithms. The normalized cuts (NCuts) algorithm [27] is a popular spectral clustering approach and has been used widely as a benchmark of image segmentation methods. Another important work in the field builds robust graphs to make use of the data similarity information [32]. Similar to NCuts, many graph-based algorithms have been proposed to utilize the information encoded in the pairwise data similarity matrix [1,17,20]. The affinity propagation (AP) [2] algorithm passes among the data the affinity message encoded in the pairwise similarity and identifies cluster centers and cluster members. In contrast to the above-mentioned algorithms, the dominant sets (DSets) algorithm [25] defines a dominant set as a graph-theoretic concept of a cluster, and extracts the clusters sequentially. Based on its nice properties, this algorithm has obtained many successful applications and further works include [11–14,24,28,29].

The DP algorithm requires as input the pairwise data distance matrix, which contains the major information needed by this algorithm. At the beginning, we

calculate the local density ρ of each data based on the distance matrix. This step can be accomplished with the cutoff kernel, Gaussian kernel or other density kernels. Then For each data, we compute its distance δ to the nearest neighbor with higher density. By intuition, the cluster centers are usually the data with large density, and they are relatively far away from each other. This implies that cluster centers are often with both large ρ and large δ. In contrast, the non-center data are usually with either small ρ or small δ. This observation enables us to isolate cluster centers from non-center data relatively easily. Another important contribution of the DP algorithm is an efficient method of grouping non-center data. It assumes that each data should be in the same cluster as its nearest neighbor with higher density. While this method has no sound theoretical foundation, it is shown to be effective in experiments.

3 Density Peak Clustering

The DP algorithm is based on the observation that cluster centers are usually high-density data surrounding by low-density ones. This observation has the following implications. First, a cluster center is the density peak in its cluster since its density is larger than those of the non-center data in its cluster. Second, a cluster center usually has a large δ as its nearest neighbor with higher density is in a different cluster. Here we see that cluster centers have both large ρ and large δ. On the contrary, the non-center data are often close to their nearest neighbor with larger density in the same cluster and their δ's are small. While the data far away from others may have large δ's, their density ρ's are usually small due to their isolation from the majority of data. In summary, the non-center data cannot have large ρ and large δ in the same time in general. This difference between cluster centers and non-center data can be utilized to identify cluster centers. In the next step, we group non-center data into clusters based on the assumption that a data has the same label as its nearest neighbor with higher density. This assumption can be supported by the observation that in a cluster non-center data have smaller density than the cluster center.

In the first step we need to calculate the local density ρ of each data. In [26] two density kernels, i.e., the cutoff kernel and Gaussian kernel, are used for this task. The cutoff kernel calculates the density of one data i as

$$\rho_i = \sum_{j \in S, j \neq i} \chi(d_c - d_{ij}), \tag{1}$$

where S denotes the set of data to be clustered, $d_c \in R$ is the cutoff distance which is determined beforehand, $d_{ij} \in R$ stands for the distance between i and j, and

$$\chi(x) = \begin{cases} 1, & x > 0, \\ 0, & x < 0. \end{cases} \tag{2}$$

Evidently, with the cutoff kernel the density is measured by the number of data in the neighborhood of radius d_c. While the cutoff kernel considers only the

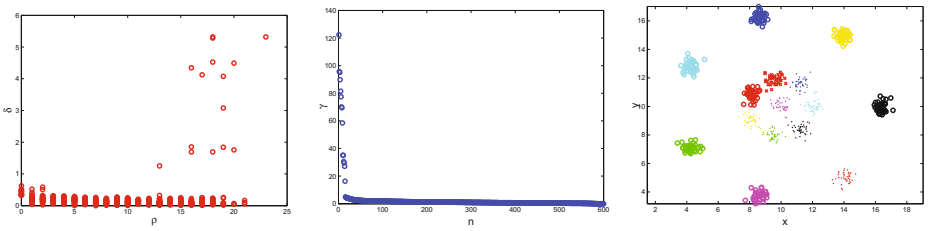
data in a neighborhood, the Gaussian kernel takes all the data into account and calculates the density by

$$\rho_i = \sum_{j \in S, j \neq i} exp(-\frac{d_{ij}^2}{d_c^2}). \tag{3}$$

Noticing that with both kernels the parameter d_c is involved, [26] proposes to determine d_c to include 1% to 2% of all data into the neighborhood on average. With the local density of each data, the distance δ_i to the nearest neighbor with higher density is calculated by definition as

$$\delta_i = \min_{j \in S, \rho_j > \rho_i} d_{ij}. \tag{4}$$

In order to identify cluster centers based on the ρ's and δ's of the data, [26] represents the data in a so-called $\rho - \delta$ decision graph. We use the R15 [30] dataset to illustrate this graph in Fig. 1, where Fig. 1(a) shows the $\rho - \delta$ decision graphs with the cutoff kernel. In this paper we determine d_c by including 1.6% of the data in the neighborhood on average. It can be observed in the $\rho - \delta$ decision graphs that there are 15 data points far away from the others. Evidently the 15 data should be regarded as cluster centers and all the others are non-center data. Since the $\rho - \delta$ decision graph involves two thresholds in identifying cluster centers, [26] further proposes to use $\gamma_i = \rho_i \delta_i$ as the single criterion and represents the data in the γ decision graph, as illustrated in Fig. 1(b). In the γ decision graph the data are sorted in the decreasing order according to γ, and those with the largest γ's will be identified as cluster centers. Finally, the clustering result is shown in Fig. 1(c), which is quite close to the ground truth.



(a) $\rho - \delta$ decision graph (b) γ decision graph (c) Clustering result

Fig. 1. The $\rho - \delta$ decision graph, γ decision graph and clustering results with the cutoff kernel on the R15 dataset.

In theory we can identify cluster centers from either the $\rho - \delta$ decision graph or γ decision graph as those data with both large ρ's and large δ's. However, in practical tasks there is no clear distinction between the "large" and "small". If we do not know the number of clusters, we are actually not sure if all the 15 data or a subset of them in Fig. 1 should be regarded as cluster centers.

In other words, while in theory the cluster centers can be identified automatically, this problem is actually not solved in [26].

Considering the difficulty in identifying cluster centers automatically, in this paper we assume that the number of clusters is determined beforehand. In this condition, we can use γ as the single criterion to select cluster centers. This method may work well in some cases, as illustrated on the R15 dataset. However, in more complex conditions, including data with very large ρ or very large δ, and different clusters have quite different density, this simple criterion may not work well. In order to improve this algorithm, in this paper we propose to normalize the δ of data, and use the normalized δ as the single criterion to identify cluster centers. In addition, we also study a new density kernel. The details of these works are presented in the next section.

4 Our Algorithm

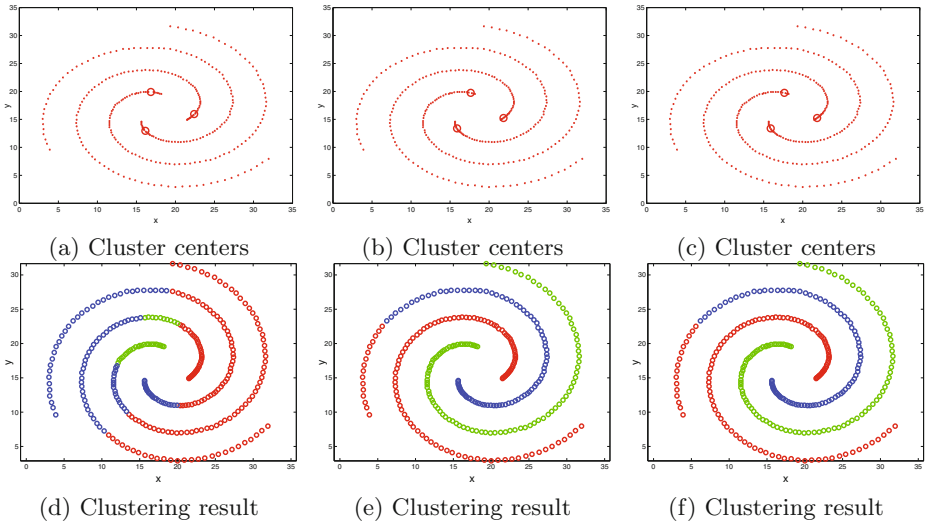
In the last section we see that on the R15 dataset, the cutoff kernel is able to generate very good clustering results, on condition that the number of clusters is given. However, this does not mean that it performs well on other datasets too. For example, we apply the cutoff kernel to the Spiral dataset [3], and show the selected cluster centers and corresponding clustering results in Fig. 2(a) and (d). On this dataset the selected cluster centers are in different clusters. However, the clustering result is not satisfactory. Our explanation of this observation is that this kernel generates unsatisfactory ρ's, δ's and γ's, and then lead to the unsatisfactory clustering result.

4.1 New Density Kernel

The cutoff kernel uses the number of clusters in a neighborhood to measure the density of one data. This kernel ignores the distance to the neighboring data and may cause information loss. In contrast, the Gaussian kernel takes all the data into account and assigns a weight to each data. Between these two extremes, we propose to measure the density by the average distance to a number of neighboring data. In implementation, we firstly calculate the average distance to five nearest neighbors, and uses the reverse of this distance as the density ρ. With this new density kernel, the cluster centers and clustering results on Spiral and Flame datasets are reported in Fig. 2(b) and (e). It is evident to see that with the new density kernel, the clustering results are improved significantly.

4.2 Distance Normalization

In the original DP algorithm we use $\gamma = \rho\delta$ as the cluster center selection criterion, and regard cluster centers as with both large ρ and large δ. However, in a cluster, it is very likely that the cluster center and its nearest neighbors are all with large density. As a result, the density ρ seems not very useful in differentiate cluster centers from non-center data.



(a) Cluster centers (b) Cluster centers (c) Cluster centers

(d) Clustering result (e) Clustering result (f) Clustering result

Fig. 2. The selected cluster centers and clustering result with different kernels and criterions on the Spiral dataset. The left column belongs to the cutoff kernel, and the middle column belongs to the new density kernel. The right column is generated with the new density kernel and normalized δ.

In this paper, we present a new criterion based on the observation that δ is more informative than ρ. We have a look at a cluster center and its neighboring data. Normally, a cluster center and its neighboring data are all with large density. Therefore *rho* is not very informative in differentiating between cluster centers and non-center data. In fact, we observe in Fig. 1 that the ρ's of all data are distributed rather evenly in the whole local density range. In contrast, only a small number of data are with large δ's, and the δ's of most data are very small. This means that δ is much more effective than ρ in differentiating between cluster centers and non-center data.

In order to improve the discriminative ability of δ further, we study the distribution of δ's of the data around the cluster centers. The cluster centers are with large δ's, and its neighbors are non-center data with smaller δ's. In contrast, the non-center data are with small δ's, and the δ's of their neighbors are usually slightly different. If we normalize the δ of one data based on the δ's of its neighboring data, the difference between cluster centers and non-center data can be enlarged. Specifically, we normalize the δ_i of each data i by

$$\delta_i' = \frac{\delta_i}{\frac{1}{|D_{inn}|}\sum\limits_{j \in D_{inn}} \delta_j}, \tag{5}$$

and then use δ' as the single criterion of cluster center selection. Evidently with this criterion the cluster centers are with large *delta′*'s, whereas those of non-center are usually small. With this new criterion, the cluster center selection and

clustering results on the Spiral and Flame datasets are shown in Fig. 2(c) and (f). By comparing with the case with only the new density kenrel, we find that the normalized δ criterion generates basically the same results as γ. This means that the normalized δ can be used as an alternative to γ in selecting cluster centers.

5 Experiments

In this section we do experiments to validate the effectiveness of the proposed algorithms. The experiments are conducted on ten datasets, including Aggregation [6], Compound [31], Pathbased [3], Spiral, D31 [30], R15 [30], Jain [16], Flame and two UCI datasets Iris and Glass. We use Normalized Mutual Information (NMI) to evaluate the clustering results.

First, we compare the new density with the cutoff and Gaussian kernels. In this part, we still use γ to select the cluster centers. The clustering result comparison among these three kernels are reported in Fig. 3. It can be observed that on the majority of datasets, our kernel performs better than or comparably to the better-performing one in the other two kernels. At the same time, we notice that our kernel is outperformed by both the cutoff and Gaussian kernels on D5 and D7. We attribute this observation to the fact the our kernel uses a fixed number of nearest neighbors to calculate local density on different datasets. This may cause unsatisfactory results in some cases, for example, the D5 and D7 datasets in our experiments. Consequently, we plan to study the possibility of an adaptive neighborhood in the future work.

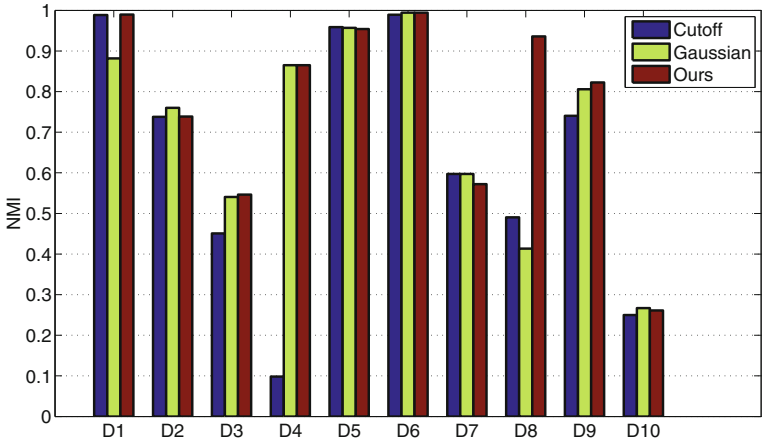


Fig. 3. The comparison of three density kernels in clustering on ten datasets. Here D1, D2, $\cdots$, D10 represent the ten datasets in the order of Aggregation, Compound, Pathbased, Spiral, D31, R15, Jain, Flame, Iris and Glass.

We then compare the whole algorithm with several other algorithms, including k-means, NCuts, DBSCAN, AP, DSets, DSets-DBSCAN [10], DP-c (DP with cutoff kernel) and DP-G (DP with Gaussian kernel). For k-means and NCuts, we set the number of clusters as ground truth. In DBSCAN, we use $MinPts = 3$ and determine Eps with the method proposed in [4]. With AP algorithm, the preference value p is determined in the following way. We firstly calculate the range of p $[p_{min}, p_{max}]$ with the code published by the authors of [2]. Then p is selected to be $p_{min} + 9.2step$, where $step = (p_{max} - p_{min})/10$. With the DSets algorithm the data similarity is calculated by $s(i,j) = exp(-d(i,j)/\sigma)$ and σ is set as $20\overline{d}$, where $d(i,j)$ refers to the Euclidean distance and $\overline{d}$ stands for the mean of all pairwise distances. The DSets-DBSCAN is a parameter free algorithm and no parameter tuning is needed. With both DP-c and DP-G, the parameter d_c is determined by including 1.6% of all data in the neighborhood on average. The comparison of these algorithms is reported in Table 1.

Table 1. Comparison of different algorithms on ten datasets with NMI.

	DSets	k-means	NCuts	DBSCAN	AP	DSets-DBSCAN	DP-c	DP-G	Ours
Aggregation	0.86	0.85	0.76	0.92	0.82	0.89	0.99	0.88	0.99
Compound	0.75	0.72	0.63	0.89	0.81	0.92	0.74	0.76	0.74
Pathbased	0.76	0.55	0.50	0.64	0.54	0.82	0.45	0.54	0.55
Spiral	0.14	0.00	0.00	0.71	0.00	0.66	0.10	0.86	0.86
D31	0.85	0.94	0.96	0.84	0.59	0.67	0.96	0.96	0.95
R15	0.83	0.92	0.99	0.87	0.74	0.91	0.99	0.99	0.99
Jain	0.43	0.37	0.33	0.73	0.46	0.87	0.60	0.60	0.57
Flame	0.60	0.40	0.42	0.83	0.57	0.90	0.49	0.41	0.94
Iris	0.65	0.76	0.74	0.75	0.79	0.60	0.74	0.81	0.82
Glass	0.31	0.37	0.37	0.40	0.33	0.37	0.25	0.27	0.26
Average	0.62	0.59	0.57	0.76	0.57	0.76	0.63	0.71	0.77

We arrive at some conclusions from Table 1. For each algorithm, the clustering results vary significantly on different datasets, and there is no one algorithm which outperform another consistently. This shows the difficulty in obtaining an universal clustering algorithm. In this case, our algorithm performs fairly evenly on different datasets, and the average clustering quality is better than those of other algorithms. This validates the effectiveness of the proposed algorithm.

As one kind of graph-based clustering algorithms, our approach requires the pairwise similarity (distance) matrix as the input, indicting a large computation load. On the basis of the original DP algorithm, our approach introduces a distance normalization step, which adds to the computation burden. Future work will be devoted to developing more efficient alternatives to the current normalization method.

In this paper we validate the proposed algorithm with a number of relatively small datasets. Noticing that the original DP algorithm has been employed in some real world problems, we believe that our improved algorithm can be applied to real world problems as well. This will be one of the directions in our future work.

6 Conclusions

On the basis of the promising density peak based clustering algorithm, in this paper we present an improved algorithm for better performance. First, we study the influence of density kernels on the clustering results, and present a new density which is based on the distance to neighboring data. Then, we investigate the criterions used in cluster center selection, and proposed to use normalized δ distance to replace the original criterion. Finally, we conduct experiments on ten datasets and compare our algorithm with eight algorithms. Experimental results validate the effectiveness of our algorithm.

Acknowledgement. This work is supported in part by the National Natural Science Foundation of China under Grant No. 61473045, and by the Natural Science Foundation of Liaoning Province under Grant Nos. 20170540013 and 20170540005.

References

1. Bello-Orgza, G., Camacho, D.: Evolutionary clustering algorithm for community detection using graph-based information. In: IEEE Congress on Evolutionary Computation, pp. 930–937 (2014)
2. Brendan, J.F., Delbert, D.: Clustering by passing messages between data points. Science **315**, 972–976 (2007)
3. Chang, H., Yeung, D.Y.: Robust path-based spectral clustering. Pattern Recogn. **41**(1), 191–203 (2008)
4. Daszykowski, M., Walczak, B., Massart, D.L.: Looking for natural patterns in data: Part 1. density-based approach. Chemometr. Intell. Lab. Syst. **56**(2), 83–92 (2001)
5. Ester, M., Kriegel, H.P., Sander, J., Xu, X.W.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: International Conference on Knowledge Discovery and Data Mining, pp. 226–231 (1996)
6. Gionis, A., Mannila, H., Tsaparas, P.: Clustering aggregation. ACM Trans. Knowl. Discov. Data **1**(1), 1–30 (2007)
7. González-Pardo, A., Granados, A., Camacho, D., de Borja Rodríguez, F.: Influence of music representation on compression-based clustering. In: IEEE Congress on Evolutionary Computation, pp. 1–8 (2010)
8. González-Pardo, A., Jung, J.J., Camacho, D.: Aco-based clustering for ego network analysis. Future Generat. Comput. Syst. **66**, 160–170 (2017)
9. González-Pardo, A., Ortíz, F.B.R., Pulido, E., Fernández, D.C.: Influence of music representation on compression-based clustering. In: ACM Workshop on Surreal Media and Virtual Cloning, pp. 9–14 (2010)

10. Hou, J., Gao, H., Li, X.: DSets-DBSCAN: a parameter-free clustering algorithm. IEEE Trans. Image Process. **25**(7), 3182–3193 (2016)
11. Hou, J., Gao, H., Xia, Q., Qi, N.: Feature combination and the kNN framework in object classification. IEEE Trans. Neural Netw. Learn. Syst. **27**(6), 1368–1378 (2016)
12. Hou, J., Liu, W., Xu, E., Cui, H.: Towards parameter-independent data clustering and image segmentation. Pattern Recogn. **60**, 25–36 (2016)
13. Hou, J., Pelillo, M.: A simple feature combination method based on dominant sets. Pattern Recogn. **46**(11), 3129–3139 (2013)
14. Hou, J., Qi, X., Qi, N.M.: Experimental study on dominant sets clustering. IET Comput. Vision **9**(2), 208–215 (2015)
15. Hou, J., Pelillo, M.: A new density kernel in density peak based clustering. In: International Conference on Pattern Recognition, pp. 463–468 (2016)
16. Jain, A.K., Law, M.H.C.: Data clustering: a user's dilemma. In: Pal, S.K., Bandyopadhyay, S., Biswas, S. (eds.) PReMI 2005. LNCS, vol. 3776, pp. 1–10. Springer, Heidelberg (2005). doi:10.1007/11590316_1
17. Menéndez, H.D., Barrero, D.F., Camacho, D.: Adaptive k-means algorithm for overlapped graph clustering. Int. J. Neural Syst. **22**(5), 1250018 (2012)
18. Menéndez, H.D., Barrero, D.F., Camacho, D.: A multi-objective genetic graph-based clustering algorithm with memory optimization. In: IEEE Congress on Evolutionary Computation, pp. 3174–3181 (2013)
19. Menéndez, H.D., Barrero, D.F., Camacho, D.: A co-evolutionary multi-objective approach for a k-adaptive graph-based clustering algorithm. In: IEEE Congress on Evolutionary Computation, pp. 2724–2731 (2014)
20. Menéndez, H.D., Barrero, D.F., Camacho, D.: A genetic graph-based approach for partitional clustering. Int. J. Neural Syst. **24**(3), 1430008 (2014)
21. Menéndez, H., Camacho, D.: A genetic graph-based clustering algorithm. In: Yin, H., Costa, J.A.F., Barreto, G. (eds.) IDEAL 2012. LNCS, vol. 7435, pp. 216–225. Springer, Heidelberg (2012). doi:10.1007/978-3-642-32639-4_27
22. Menéndez, H.D., Otero, F.E.B., Camacho, D.: MACOC: a medoid-based ACO clustering algorithm. In: Dorigo, M., Birattari, M., Garnier, S., Hamann, H., Montes de Oca, M., Solnon, C., Stützle, T. (eds.) ANTS 2014. LNCS, vol. 8667, pp. 122–133. Springer, Cham (2014). doi:10.1007/978-3-319-09952-1_11
23. Menéndez, H.D., Plaza, L., Camacho, D.: Combining graph connectivity and genetic clustering to improve biomedical summarization. In: IEEE Congress on Evolutionary Computation, pp. 2740–2747 (2014)
24. Zemene, E., Pelillo, M.: Interactive image segmentation using constrained dominant sets. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9912, pp. 278–294. Springer, Cham (2016). doi:10.1007/978-3-319-46484-8_17
25. Pavan, M., Pelillo, M.: Dominant sets and pairwise clustering. IEEE Trans. Pattern Anal. Mach. Intell. **29**(1), 167–172 (2007)
26. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science **344**, 1492–1496 (2014)
27. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 167–172 (2000)
28. Tripodi, R., Pelillo, M.: A game-theoretic approach to word sense disambiguation. Comput. Linguist. **43**(1), 31–70 (2017)
29. Vascon, S., Mequanint, E.Z., Cristani, M., Hung, H., Pelillo, M., Murino, V.: Detecting conversational groups in images and sequences: a robust game-theoretic approach. Comput. Vis. Image Underst. **143**, 11–24 (2016)

30. Veenman, C.J., Reinders, M., Backer, E.: A maximum variance cluster algorithm. IEEE Trans. Pattern Anal. Mach. Intell. **24**(9), 1273–1280 (2002)
31. Zahn, C.T.: Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans. Comput. **20**(1), 68–86 (1971)
32. Zhu, X., Loy, C.C., Gong, S.: Constructing robust affinity graphs for spectral clustering. In: IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1450–1457 (2014)