ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
DOI: 10.17559/TV-20160531025240

SEMI-SUPERVISED AFFINITY PROPAGATION BASED ON DENSITY PEAKS

Limin Wang, Xing Tao, Xuming Han, Jialing Han, Ying Liu, Guangyu Mu

Original scientific paper

In view of the unsatisfying clustering effect of affinity propagation (AP) clustering algorithm when dealing with data sets of complex structures, a semi-supervised affinity propagation clustering algorithm based on density peaks (SAP-DP) was proposed in this paper. The algorithm uses a new algorithm of density peaks (DP) which has the advantage of the manifold clustering with the idea of semi-supervised, builds pairwise constraints to adjust the similarity matrix, and then executes the AP clustering. The results of the simulation experiments validated that the proposed algorithm has better clustering performance compared with conventional AP.

Keywords: *Affinity Propagation; Density Peaks; pairwise constraints; semi-supervised learning*

Polu-nadzirana propagacija afiniteta temeljena na vršnoj gustoći

Izvorni znanstveni članak

Zbog nezadovoljavajućeg učinka grupiranja (klasteriranja) pomoću algoritma grupiranja propagacijom afiniteta (AP - affinity propagation) u slučaju nizova podataka složene strukture, u radu se predlaže polu nadzirani algoritam grupiranja propagacije afiniteta temeljen na vršnoj gustoći (SAP-DP). Taj algoritam primjenjuje novi algoritam vršne gustoće (DP - density peaks) čija je prednost višestruko grupiranje uz polu-nadziranje, izgradnja udvojenih ograničenja zbog usklađivanja s matricom sličnosti, a zatim izvršenje grupiranja propagacijom afiniteta. Rezultati simulacijskih eksperimenata potvrdili su da je grupiranje predloženim algoritmom učinkovitije od grupiranja konvencionalnom propagacijom afiniteta (AP).

Ključne riječi: *polu-nadgledano učenje; propagacija afiniteta; vršna gustoća; udvojena ograničenja*

1 Introduction

Affinity Propagation clustering (AP) is a quite different and efficient clustering algorithm. It simultaneously considers all data points as potential exemplars, and it does not require the number of clusters to be predetermined like other clustering algorithms do [1]. Affinity propagation clustering algorithm was published by Brendan J. Frey and Delbert Dueck in the *Science* in 2007.

In recent years, scholars have developed many improving methods, all focus on three issues: application study, similarity matrix and complex data processing.

Now Affinity Propagation is widely used in text segmentation [2], artificial immune system [3, 4], image recognition [5] and many other fields [6÷8]. Canadian scholars Hassanabadi et al. [9] present a novel, mobility-based clustering scheme for Vehicular Ad hoc Networks, which forms clusters using the Affinity Propagation algorithm in a distributed manner. Austrian scholars Bodenhofer U. et al. [10] provided an R implementation of AP algorithm to account for the ubiquity of *R* in bioinformatics. By introducing an idea of the emoticon to AP, Zhang Lumin et al. [11] proposed a novel approach to mine online events based on emoticons. Lu Weiming et al. [12] proposed a distributed AP clustering algorithm based on MapReduce to effectively address the large scale data.

Delbert Dueck and Brendan J. Frey [26] use nonmetric similarity to increase accurate rates of imagines classification. Zhengdong Lu [27] proposed two kinds of affinity information changing the matrix to yield better results with fewer constraints. Zhen Zhang [28] proposes STI-AP with defined manifold similarity and semi-supervised learning to reduce the complexity of marking sampled flows. They improve calculation methods of similarity in order to be suitable to some special problem. Similarity is close relation to data structures, different data need different methods.

Jianpeng Zhang [29] takes an improved weighted and hierarchical affinity propagation to reconstruct the AP models when detecting a new emerging class model. Streaming data have characteristics of dynamic distribution, Zhang X., Furtlehner C. and Germain-Renaud C. [30] apply affinity propagation as efficient clustering algorithm to VANETs. Like other algorithms, AP also has its limitations, and not a single method can be suitable for all problems, facing complex data such as manifold, we need to improve it.

The AP works based on similarities, considers all the data points as the potential cluster centres, and then through iterative competition to obtain the optimal clustering results. AP is different from other clustering algorithms. It does not need to specify the clustering number which quickly and efficiently deals with the large-scale data. But because of lacking prior information the algorithm would create many local clusters when processing complex data. According to the above problem, the paper puts forward the algorithm Semi-supervised Affinity Propagation based on Destiny Peaks (SAP-DP).

2 Affinity Propagation

Affinity propagation is a new and efficient algorithm which is based on similarities between pairs of data points and considers all data points as the potential clustering centre. Real-valued messages are exchanged between data points until a high quality set of exemplars and corresponding clusters gradually emerges. Because of its simplicity, general applicability, and performance, we believe affinity propagation will prove to be of broad value in science and engineering [25].

Affinity propagation takes as input a collection of real-valued similarities between data points, where the similarity $s(i, k)$ indicates how well the data point with index k is suited to be the exemplar for data point i [1].

Each similarity is set to a negative squared error (Euclidean distance): for points *i* and *k* the similarity is:

$$s(i,k) = -d_{ik} = -\left\|x_i - x_k\right\|. \quad (1)$$

A priori, all data points are taken as the potential cluster centres. A data point with large value of s(*k*, *k*) is more likely chosen as exemplar.

These values are referred to as preference parameters; they play important roles in determining the number of exemplars. Initially all data points are equally suitable as exemplars, the preference parameter should be set to a common value *p*− this value can be varied to produce different numbers of clusters. In most cases, this shared value could be the median of the input similarities.

$$p = median(s(:)). \quad (2)$$

During the iteration, there are two types of messages exchanged between data points, and each takes into account a different kind of competition. Messages can be combined at any stage to decide which points are exemplars and, for every other point, which exemplar it belongs to. Fig. 2 shows affinity propagation illustrated for two-dimensional data points, where negative Euclidean distance was used to measure similarity. Each point is coloured according to the current evidence that it is a cluster centre (exemplar). The darkness of the arrow directed from point *i* to point *k* corresponds to the strength of the transmitted message that point *i* belongs to exemplar point *k* [1].

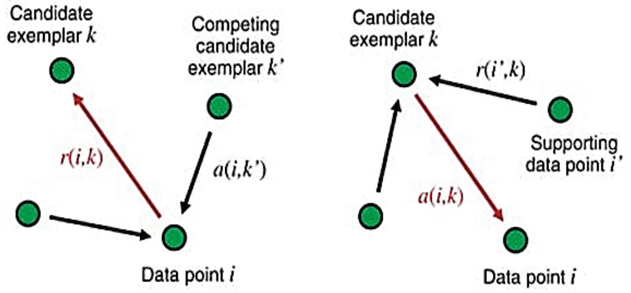

Figure 1 How affinity propagation works

The core of AP is mutual transfer of the two pieces of information. The "responsibility" *r*(*i*,*k*) from point *i* to point *k*. It reflects how well-suited point *k* is to serve as the exemplar for point *i*. The "availability" *a*(*i*,*k*) from point *k* to point *i*. It reflects how appropriate it would be for point *i* to choose point *k* as its exemplar. From the viewpoint of evidence, the larger the *r*(:,*k*)+*a*(:,*k*), the more probability the point *k* has as a final cluster centre.

$$r(i,k) \leftarrow s(i,k) - \max_{k' \neq k}\left\{a\left(i,k'\right) + s\left(i,k'\right)\right\}, \quad (3)$$

$$a(i,k) \leftarrow \left\{\begin{array}{ll} \min\left\{0, r(k,k) + \sum\limits_{i' \notin \{i,k\}} \max\left\{0, r(i'+k)\right\}\right\} & i \neq k \\ \sum\limits_{i' \neq k} \max\left\{0, r(i',k)\right\} & i = k \end{array}\right\}, \quad (4)$$

In order to avoid oscillation, AP introduces damping factor ($\lambda \in [0,\ 1)$) to information update. This paper selects lambda 0,5. *t* is the iteration. And each iteration of affinity propagation consisted of (i) updating all responsibilities given the availabilities, (ii) updating all availabilities given the responsibilities, and (iii) combining availabilities and responsibilities to monitor the exemplar decisions and terminate the algorithm when these decisions did not change for 10 iterations.

$$r^{(t+1)}(i,k) \leftarrow (1-\lambda) r^{(t+1)}(i,k) + \lambda r^{(t)}(i,k), \quad (5)$$

$$a^{(t+1)}(i,k) \leftarrow (1-\lambda) a^{(t+1)}(i,k) + \lambda a^{(t)}(i,k), \quad (6)$$

A decision matrix *E* is calculated after each update. Decision matrix *E* represents whether point *i* chooses point *k* as its exemplar or not.

$$E(k) = \arg\max_{k}\left(a\left(i,k\right) + r\left(i,k\right)\right). \quad (7)$$

3 Clustering by fast search and find of Density Peaks (DP)

The DP algorithm has its basis only in the distance between data points. It is able to detect nonspherical clusters and to automatically find the correct number of clusters [13]. DP algorithm has two quantities: for each point *i*, its local density ρ_{i} and its distance δ_{i} from points of higher density. Both these quantities depend only on the distances between data points, which are assumed to satisfy the triangular inequality. The local density ρ_{i} of data point *i* is defined as:

$$\rho_{\mathrm{i}} = \sum_j \chi(d_{ij} - d_c). \quad (8)$$

where $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise, and d_{c} is a cut-off distance. Basically, ρ_i is equal to the number of points that are closer than d_{c} to point *i*.

δ_i is measured by computing the minimum distance between the point *i* and any other point with higher density. For the point with the highest density, it is taken $\delta_i = \max_j(d_{\mathrm{ij}})$. Generally, one can choose d_{c} so that the average number of neighbours (τ) is around 1 to 2 % of the total number of points in the data set. DP chooses the only points of high δ_i and relatively high ρ_i are the cluster centres. After the cluster centres have been found, each remaining point is assigned to the same cluster as its nearest neighbour of higher density. Detailed calculation methods follow these formulas:

$$\chi(x) = \begin{cases} 1 & if \quad x < 0 \\ 0 & if\ otherwise \end{cases}, \quad (9)$$

$$\delta_i = \left\{\begin{array}{ll} \max\limits_{j}(d_{ij}) & \rho = \max\left(\rho_1, \rho_2, ..., \rho_n\right) \\ \min\limits_{j:\rho_j > \rho_i}(d_{ij}) & otherwise \end{array}\right\}. \quad (10)$$

This observation, which is the core of the algorithm, is illustrated by the simple example in Fig. 2 and Fig. 3.

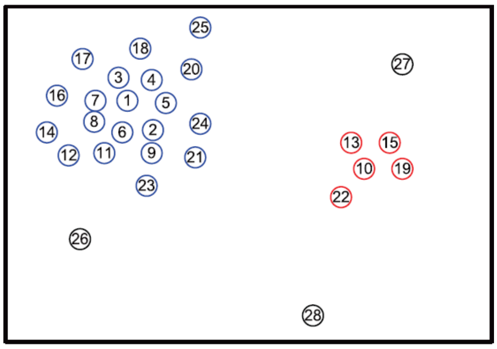

Figure 2 Point distribution in two dimensions

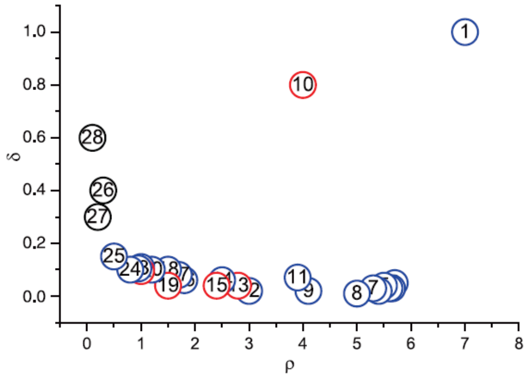

Figure 3 Decision graph in two dimensions

Fig. 2 shows 28 points embedded in a two-dimensional space. The density maxima are at points 1 and 10 so they are identified as cluster centres. Fig. 3 shows the plot of δ_i as a function of ρ_i for each point. The value of δ for point 9 and 10, with similar value of ρ, is very different: Point 9 belongs to the cluster of point 1, and several other points with a higher ρ are very close to it, whereas the nearest neighbour of higher density of point 10 belongs to another cluster. Hence, as anticipated, the only points of high δ and relatively high ρ are the cluster centres. Points 26, 27 and 28 have a relatively high δ and a low ρ because they are isolated; they can be considered as cluster composed of a single point, namely, outliers [13].

4 Semi-supervised clustering

There is a new method of semi-supervised clustering based on AP algorithm [14]. The algorithm has two kinds of pairwise constraints, must-link, where the two data points must belong to the same cluster, i.e. $M=\{(x_i, x_j)\}$, and cannot-link, where two data points should not be in the same cluster, i.e. $C=\{(x_i, x_j)\}$ [2]. The detailed rules for updating the matrix are as follows.

Step 1: For the data point pairs in prior information that meet the must-link constraint and the data point pairs newly accord with the must-link constraint after logical extension, perform similarity update as below.

$$\left(x_i,x_k\right)\in M \Rightarrow s\left(i,j\right)=0\ \&\ s\left(j,i\right)=0, \tag{11}$$

$$(x_i,x_k)\notin M\ \&\ (x_i,x_j)\in M\ \&\ (x_j,x_k)\in M \Rightarrow (x_i,x_k)\in M, \tag{12}$$

Step 2: For the data point pairs in prior information that meet the cannot-link constraint, perform similarity update as below.

$$(x_i,x_j)\in C \Rightarrow s(i,j)=-\infty\ \&\ s(j,i)=-\infty. \tag{13}$$

Step 3: Perform global adjustment to the unknown data points based on the principle of the shortest path according to the results of steps 1 and 2. If there is a data point that connects to both data points in a data point pair pending for adjustment, and the sum of the similarities between this data point and the two data points in the pair is greater than the similarity of the data point pair, update the similarity of the data point pair to the sum.

$$\begin{aligned}&(x_i,x_j)\notin\{M\cup C\}\Rightarrow\\&s(i,j)=\max\{s(i,j),s(i,k)+s(k,j)\}.\end{aligned} \tag{14}$$

5 Semi-supervised Affinity Propagation based on Density Peaks

The paper randomly selects 80 % of the data set as the training data, and chooses the reasonable average number of neighbours (τ). Firstly through DP cluster to gain constraint information, and then update the similarity matrix by semi-supervised clustering, finally use AP to calculate the result. Here is the process of the proposed algorithm: step 1 is to initialize $r(i, k)$, $a(i, k)=0$, $\lambda=0.5$ and t = 10000; step 2 is to randomly choose 80 % of the data set to perform DP clustering. Step 3 is to gain the pairwise constraints which have the musk-link and cannot-link. Step 4 is to update the similarity matrix according to the pairwise constraints. Step 5 is to perform AP clustering and then use the Silhouette index and F-Measure index to test the result. Fig. 4 illustrates detailed flowchart of the clustering processes.

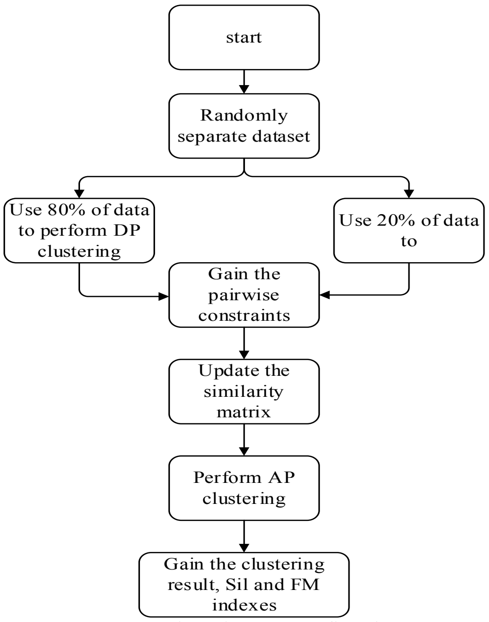
start
Randomly separate dataset
Use 80% of data to perform DP clustering
Use 20% of data to
Gain the pairwise constraints
Update the similarity matrix
Perform AP clustering
Gain the clustering result, Sil and FM indexes

Figure 4 Clustering process flowchart

6 Experimental results

We present a set of clustering experiments on many datasets, including three synthetic datasets, three UCI datasets, as shown in Tab. 1. All experiments were performed with MATLAB 2012b on a computer with Inter(R) Pentium 2.9 GHz processer, 4GB RAM, 500GB hard drive.

Table 1 Experimental datasets

Datasets	Clustering number		
	Number of samples	Dimensions	True number
Iris	150	4	3
Seeds	210	7	3
Heart	303	13	2
Spiral	312	2	3
Flame	240	2	2
Aggregation	788	2	7

6.1 Silhouette (Sil) index

Assume a data set with n samples be divided into k clusters C_i (i =1, 2, …, k), $a(t)$ is the average dissimilarity of sample t in C_j to all other samples in C_j, $d(t, C_i)$ is the average dissimilarity of sample t in C_j to all samples in another cluster C_i, then $b(t) = \min\{d(t, C_i)\}$, i = 1, 2, …, k, $i{\neq}j$. The formula to calculate the Silhouette index Sil of sample t is:

$$Sil\left(t\right)=\frac{\left[b\left(t\right)-a\left(t\right)\right]}{\max\left\{a\left(t\right),b\left(t\right)\right\}}. \tag{15}$$

6.2 F-measure (FM) index

F-measure measures the grammar's accuracy. It considers both the precision P and the recall R of the algorithm: P is the ratio of the number of correct results to the number of all returned results, and R is the ratio of the number of correct results to the number of results that should have been returned. P, R and F-measure (F) are defined as follows.

$$P=\left(M_j,C_i\right)=\frac{\left|M_j\cap C_i\right|}{\left|C_i\right|}, \tag{16}$$

$$R=\left(M_j,C_i\right)=\frac{\left|M_j\cap C_i\right|}{\left|M_j\right|}, \tag{17}$$

$$F\left(M_j,C_i\right)=\frac{2\cdot P\left(M_j,C_i\right)\cdot R\left(M_j,C_i\right)}{P\left(M_j,C_i\right)+R\left(M_j,C_i\right)}, \tag{18}$$

$$F=\sum_j\frac{\left|M_j\right|}{N}\max_i F\left(M_j,C_i\right). \tag{19}$$

6.3 Comparison and analysis of the results

We compared the performance of the proposed algorithm with AP on three synthetic datasets and three UCI datasets. We tested the Silhouette index and F-measure index of the three algorithms based on the true clustering number. The result is shown as follows.

From Tab. 2 and Fig. 5 we can see that the clustering accuracy of the proposed SAP-DP algorithm is better than two other algorithms which is shown from the FM index. As for the clustering quality which is shown from the Silhouette index, the SAP-DP get better result in dataset Iris, Heart and Aggregation while it is poor at the datasets spiral, seeds and flame. It indicates that the Silhouette index is sensitive to the spherical data. The result of the Silhouette index proves that the SAP-DP can effectively construct a similarity matrix, improve the compactness of within-class and the separability of inter-class while KMeans and AP can only recognize the spherical data, so the clustering trend is more obvious. The result of the F-measure index shows that the Clustering accuracy of SAP-DP is improved obviously. In order to intuitively compare the three algorithms, we choose three synthetic datasets to plot the Decision graphs and Clustering results.

Table 2 Comparison of clustering index

Datasets	KMeans		AP		SAP-DP	
	Sil	*FM*	*Sil*	*FM*	*Sil*	*FM*
Iris	0,6244	0,6580	0,6784	0,8064	0,7304	0,8320
Seeds	0,6473	0,8067	0,6263	0,7885	0,6307	0,8315
Heart	0,3195	0,5149	0,3187	0,5126	0,3198	0,6500
Aggregation	0,6058	0,7742	0,1623	0,5958	0,6345	0,9686
Spiral	0,3274	0,5317	0,4998	0,2891	-0,0733	0,8108
Flame	0,5333	0,7363	0,4660	0,5944	0,4399	0,9463

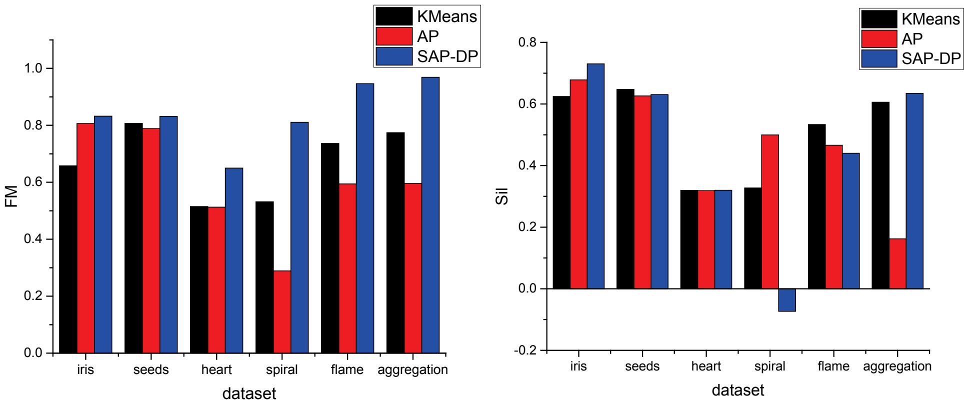

Figure 5 Comparison of clustering quality and accuracy

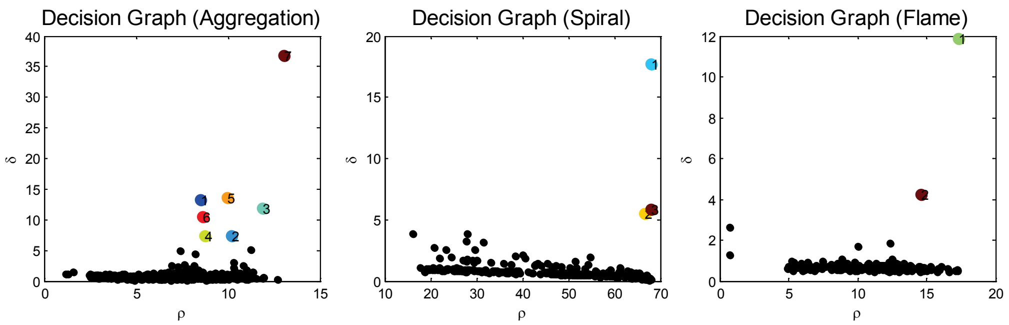

Figure 6 Decision graph on Aggregation, Spiral and Flame datasets

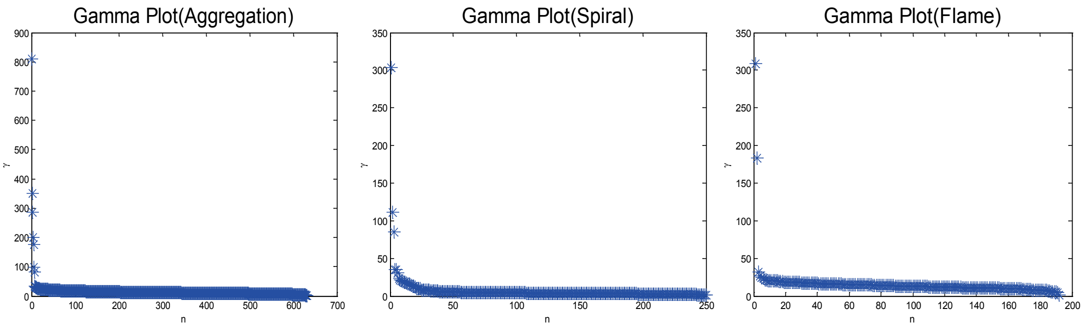

Figure 7 The value of gamma in decreasing order for datasets

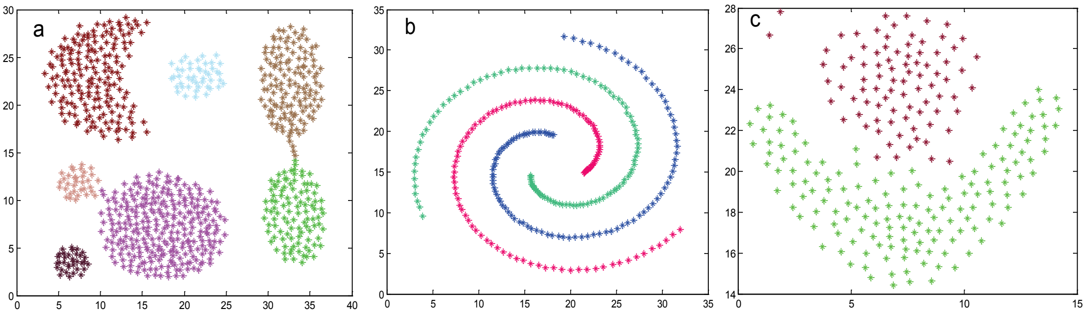

Figure 8 Three synthetic datasets: (a) Aggregation dataset (b) Spiral dataset (c) Flame dataset

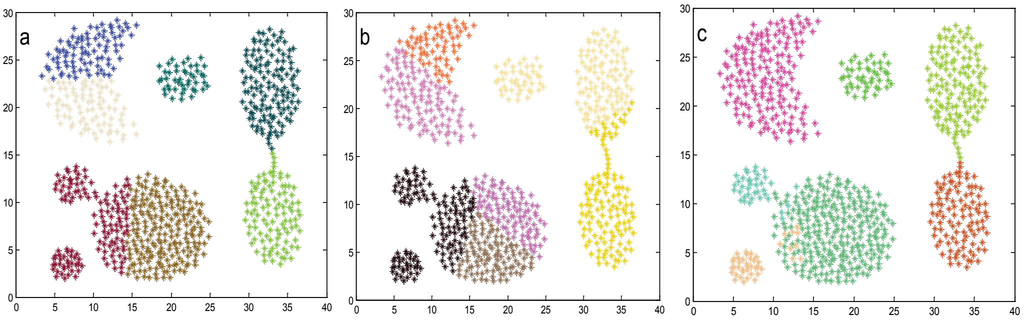

Figure 9 Clustering on Aggregation dataset by KMeans (a), AP (b) and SAP-DP (c)

Fig. 6 shows the decision graphs of three datasets. Based on the principle of DP algorithm, we choose the points with high δ and relatively high ρ as the cluster centres. The better centre we choose, the more accurate semi-supervised information we gain.

Fig. 7 shows the value of gamma (γ) in decreasing order for three synthetic datasets. It provides the evidence for choosing the number of clustering centre. For example, the graph of aggregation's gamma shows that the quantity starts growing anomalously below a rank order number 7. Therefore, we performed the analysis by using 7 centres.

$$\gamma_i = \rho_i * \delta_i . \qquad (20)$$

In Fig. 9 one can see: The aggregation is a composite spherical dataset which has a complex structure. The original AP and KMeans are hard to gain the right clustering while SAP-DP can get a more reasonable clustering. The F-Measure index of AP is 0,5958 and KMeans's is 0,7742 while the SAP-DP's is 0,9686. The Silhouette index of AP is 0,1623 and KMeans's is 0,6058 while the SAP-DP's is 0,6345. Therefore, the proposed algorithm can process composite spherical data more efficiently.

In Figs. 10 and 11 one can see: Original AP and KMeans are hard to gain the true cluster number. When we adjust the preference to the true cluster, the F-Measure index of SAP-DP is higher than the AP and KMeans, while the Silhouette index of SAP-DP is lower than the other two algorithms. Silhouette index is sensitive to the spherical data while it has a bad result in the nonspherical data. So Figs. 5 and 6 prove the proposed algorithm has the better clustering ability on nonspherical data.

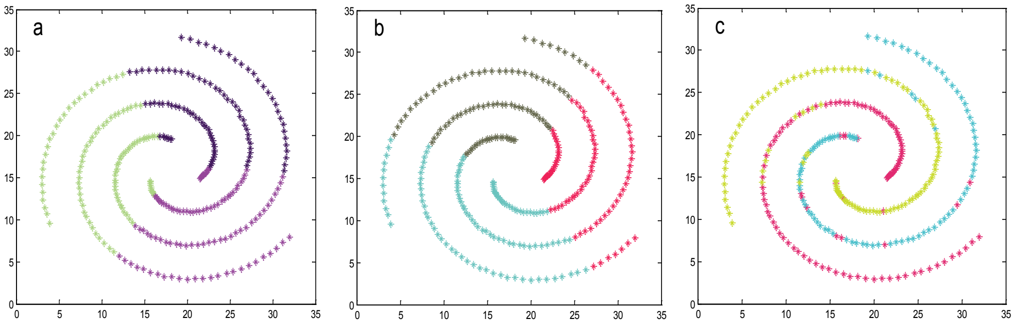

Figure 10 Clustering on Spiral dataset by KMeans(a), AP (b) and SAP-DP (c)

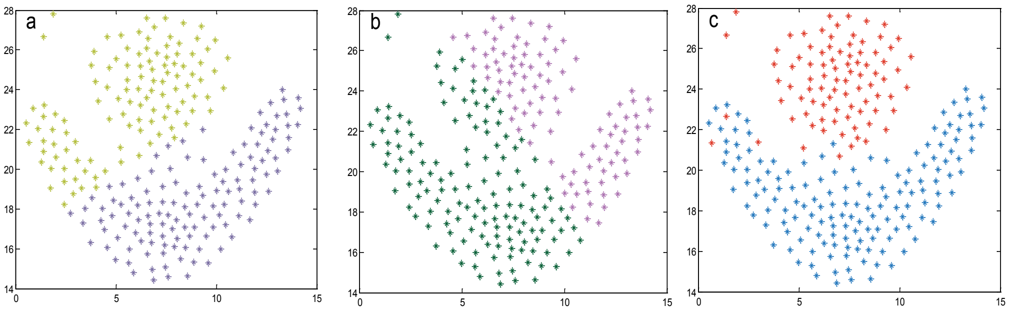

Figure 11 Clustering on Flame dataset by KMeans(a), AP (b) and SAP-DP (c)

Table 3 The seismic data

Date	Location	Magnitude	Richter magnitude	Epicentral intensity	Earthquake victim	Death toll	Total casualties	Direct economic loss
19951024	Yunnan Wuding	1	6	8	101,4	58	13 815	74 383
20000115	Yunnan Yaoan	1	6,5	8	69,4	7	2528	106 621
20010412	Yunnan Shidian	1	5,9	8	66,9648	3	235	50 490
20030224	Xinjiang Bachu	1	6,8	9	65,9392	268	4853	139 792
20031025	Gansu Minle	1	6,1	8	19,0101	10	46	50 140
20030816	Inner Mongolia Balin	1	5,9	8	48,0869	4	1064	80 649
20030721	Yunnan Dayao	1	6,2	8	32,2962	16	793	59 190
19950712	Yunnan Menglian	2	7	6	57,7	11	136	20 550
20031115	Yunnan Ludian	2	5,1	7	23,6652	4	94	19 190
20041019	Yunnan Baoshan	2	5	6	39,8327	0	16	21 720
20040324	Inner Mongolia Wuzhu	2	5,9	7	23,6652	4	94	19 190
20040810	Yunnan Ludian	2	5,6	8	31,3556	4	597	33 226
19960319	Xinjiang Ashitu	2	7	6	30	24	128	35 366,81
19970121	Xinjiang Jiashi	2	6	6	23,82	12	44	37 413,6
19970411	Xinjiang Jiashi	2	7	7	25,34	8	62	32 003
19980110	Hebei Zhangbei	2	6	8	16,9	49	11 439	78 800
19980827	Xinjiang Jiashi	2	7	6,5	12,05	3	13	12 507,85
19981119	Yunnan Ninglang	2	6	6	12,8	5	1487	39 114
19950722	Gansu Yongdeng	3	6	7	22,4	10	592	6779
19960925	Yunnan Lijiang	3	6	7	3,9	1	141	3080
19961221	Sichuan Batang	3	6	7	1,1	2	60	3998,5
19970301	Xinjiang Jiashi	3	6	7	13,1	1	6	8819,48
19990311	Hebei Zhangbei	3	5,6	7	1,2769	0	3	1105,5
19990415	Gansu Wenxian	3	4,7	6	8,1339	1	30	627,1
20000429	Henan Neixiang	3	4,7	6	19	1	28	5680
20010312	Henan Lancang	3	5	6	4,5449	0	6	5575
20010608	Yunnan Shidian	3	5,3	6	10,6128	1	15	3660
20021020	Inner Mongolia Wuzhu	3	5	6	2,7	0	0	800,76

7 Application of SAP-DP on seismic analysis

We chose the seismic data from China Earthquake Data Centre to test the feasibility of the proposed algorithm. There are six measurement indexes used in the test: Richter magnitude, epicentral intensity, earthquake victim, death toll, total casualties and direct economic loss (Tab. 3). The earthquake disasters magnitude in China is divided into: general, moderate, severe and catastrophic. Because of the rareness of the catastrophic earthquakes, we chose the previous three magnitudes to test the data. We respectively defined "general", "moderate", "severe" as "3", "2", "1".

Table 4 Result of the SAP-DP clustering

Date	Location	Magnitude (actual)	Magnitude (test)
19951024	Yunnan Wuding	1	1
20000115	Yunnan Yaoan	1	1
20010412	Yunnan Shidian	1	1
20030224	Xinjiang Bachu	1	1
20031025	Gansu minle	1	1
20030816	Inner Mongolia Balin	1	1
20030721	Yunnan Dayao	1	1
19950712	Yunnan menglian	2	2
20031115	Yunnan Ludian	2	2
20041019	Yunnan Baoshan	2	2
20040324	Inner Mongolia Wuzhu	2	2
20040810	Yunnan Ludian	2	2
19960319	Xinjiang Ashitu	2	2
19970121	Xinjiang Jiashi	2	2
19970411	Xinjiang Jiashi	2	2
19980110	Hebei Zhangbei	2	1
19980827	Xinjiang Jiashi	2	3
19981119	Yunnan Ninglang	2	2
19950722	Gansu Yongdeng	3	3
19960925	Yunnan Lijiang	3	3
19961221	Sichuan Batang	3	3
19970301	Xinjiang Jiashi	3	3
19990311	Hebei Zhangbei	3	3
19990415	Gansu Wenxian	3	3
20000429	Henan Henei	3	3
20010312	Henan Lancang	3	3
20010608	Yunnan Shidian	3	3
20021020	Inner Mongolia Wuzhu	3	3

As seen in Tab. 4, the actual magnitude and test magnitude are basically identical. There are two samples that are misestimated. The F-Measure index of the result is 0,85. It shows that the application of the proposed algorithm on seismic analysis is effective. It provides a relatively effective research tool for the earthquake classification field.

8 Conclusion

For the incapability of affinity propagation clustering algorithm to produce ideal clustering results when dealing with nonspherical data, a novel semi-supervised affinity propagation clustering algorithm based on density peaks was proposed in this paper. The proposed SAP-DP algorithm makes full use of the manifold clustering features of the density peak, accurately identifies the potential manifold structure of complicated data, introduces the idea of a semi-supervised learning, and builds pairwise constraint condition by clustering. The pairwise constraints are used to update the similarity matrix that reflects the relationship of similarity more reasonable. Then the algorithm is executed to reach the result. Taken together, compared with the traditional AP algorithm, the Semi-supervised Affinity Propagation based on Density Peaks has better accuracy and performance.

Acknowledgments

The research was supported by the National Science Foundation of China under grant No. 61202306, 61472049, 61402193, 61572225 and 61202309; the National Society Science Foundation of China under grant No. 15BGL090; the Foundation of Jilin Provincial Science & Technology Department under Grant No. 201215119, 20130522177JH, 20130101072JC, 20150312001ZG and 20160101259JC; the Society Science Research Foundation of Jilin Province under grant No. 2014B166; Jilin Province Education Department under grant No. 2014159 and 2016 [78]; the Open Fund Project of Logistics Industry Economy and the Intelligent Logistics Laboratory of Jilin Province under grant No. 02060311 and 02060312; the Fund of Jilin Province Economic Structure Strategic Adjustment under grant No. 2015Y040 and 2015Y037.

9 References

[1] Brendan, J. Frey; Delbert Dueck. Clustering by Passing Messages between Data Points. // Science. 315, 5814(2007), pp. 972-976. DOI: 10.1126/science.1136800

[2] Kazantseva; Anna; Stan Szpakowicz. Linear Text Segmentation Using Affinity Propagation. // Proceedings of the Conference on Empirical Methods in Natural Language Processing / Association for Computational Linguistics, 2011, pp. 284-293.

[3] Chu Yuezhong; Xu Bo. Design of Classifier Based on Combination of Artificial Immune System and AP Algorithm. // Journal of Nanjing University of Aeronautics & Astronautics, 45, 2(2013), pp. 232-238.

[4] Zhu Sifeng; Liu Fang. Application of Immune Clustering Algorithm to the Analysis of Gene Expression Data. // Journal of Beijing University of Posts and Telecommunications, 33, 2(2010), pp. 54-57.

[5] Su Hongjun; Sheng Yehua. Adaptive Affinity Propagation with Spectral Angle Mapper for Semi-supervised Hyperspectral Band Selection. // Appl Opt. 51, 14(2012), pp. 2656-2663. DOI: 10.1364/AO.51.002656

[6] Borile C. Using affinity propagation for identifying subspecies among clonal organisms: lessons from M. tuberculosis. // BMC bioinformatics, 12, 1(2011), pp. 224. DOI: 10.1186/1471-2105-12-224

[7] Zhu Sifeng; Liu Fang. Application of Immune Clustering Algorithm to the Analysis of Gene Expression Data. // Journal of Beijing University of Posts and Telecommunications, 33, 2(2010), pp. 54-57.

[8] Sakellariou A; Sanoudou D. Combining Multiple Hypothesis Testing and Affinity Propagation Clustering Leads to Accurate, Robust and Sample Size Independent Classification on Gene Expression Data. // BMC bioinformatics, 13, 1(2012), pp. 270. DOI: 10.1186/1471-2105-13-270

[9] Hassanabadi B; Shea C. Clustering in Vehicular Ad Hoc Networks Using Affinity Propagation. // Ad Hoc Networks, 13, 1(2014), pp. 535-548. DOI: 10.1016/j.adhoc.2013.10.005

[10] Bodenhofer U; Kothmeier A. APCluster: an R package for affinity propagation clustering. // Bioinformatics, 27, 17(2011), pp. 2463-2464. DOI: 10.1093/bioinformatics/btr406

[11] Zhang Lumin; Jia Yan. Online Bursty Events Detection Based on Emotions. // Chinese Journal of Computers, 36, 8(2013), pp. 1659-1667. DOI: 10.3724/SP.J.1016.2013.01659

[12] Lu Weiming; Du Chenyang. Distributed Affinity Propagation Clustering Based on MapReduce. // Journal of Computer Research and Development, 49, 8(2012), pp. 1762-1772.

[13] Rodriguez A; Laio A. Clustering by Fast Search and Find of Density Peaks. // Science, 344, 6191(2014), pp. 1492-1496. DOI: 10.1126/science.1242072

[14] Xiao Yu; Yu Jian. Semi-supervised Clustering Based on Affinity Propagation Algorithm. // Journal of Software, 19, 11(2008), pp. 2803-2813, 2008.

[15] Givoni, I.; Chung, C. Hierarchical Affinity Propagation. // arXiv preprint arXiv, 1202, 3722(2012), pp. 238-246.

[16] Shang, F. Fast Affinity Propagation Clustering: a Multilevel Approach. // Pattern recognition, 45, 1(2012), pp. 474-486. DOI: 10.1016/j.patcog.2011.04.032

[17] Fujiwara, Y.; Irie, G. Fast Algorithm for Affinity Propagation. // Proceedings-International Joint Conference on Artificial Intelligence / 22, 3(2011), pp. 2238.

[18] Anand, S. Semi-supervised Kernel Mean Shift Clustering. // IEEE Trans Pattern Anal Mach Intell. 36, 6(2013), pp. 1201-1215. DOI: 10.1109/TPAMI.2013.190

[19] Yan Yang; Chen L. Fuzzy Semi-supervised Co-clustering for Text Documents. // Fuzzy Sets & Systems, 215, 3(2013), pp. 74-89.

[20] Xiong S; Azimi, J. Active Learning of Constraints for Semi-supervised Clustering. // Knowledge & Data Engineering IEEE Transactions on, 26, 1(2014), pp. 43-54. DOI: 10.1109/TKDE.2013.22

[21] Zhang Zhen; Wang Binqiang. Semi-supervised Affinity Propagation Clustering Algorithm Based on Stratified Combination. // Journal of Electronics & Information Technology, 35, 3(2013), pp. 645-651.

[22] Feng Xiaolei; Yu Hongtao. Semi-supervised Affinity Propagation Clustering Based on Manifold Distance. // Application Research of Computers, 28, 10(2011), pp. 3656-3664.

[23] Sheng Jiagen; Liu Sifeng. Grey Relational Analysis for Fuzzy Clustering. // Journal of the China Society for Scientific and Technical Information, 29, 3(2010), pp. 493-496.

[24] Hu Rui; Lin Zhaowen. DataStreams Clustering Algorithm Based on Density and Sliding Window. // Computer Science, 38, 5(2011), pp. 145-148.

[25] Napoleon, D.; Baskar, G. An Efficient Clustering Technique for Message Passing between Data Points using Affinity Propagation. // International Journal on Computer Science and Engineering, 3, 1(2011), pp. 8-14.

[26] Dueck, D.; Frey, B. Non-metric affinity propagation for unsupervised image categorization. // IEEE 11th International Conference on Computer Vision / IEEE, 2007, pp. 1-8.

[27] Lu, Z.; Carreira-Perpinan, M. Constrained spectral clustering through affinity propagation. // IEEE Conference on Computer Vision and Pattern Recognition / IEEE, 2008, pp. 1-8.

[28] ZHANG Zhen; WANG Bin-qiang; LI Xiang-tao. Semi-supervised Traffic Identification Based on Affinity Propagation. // Acta Optica Sinica, 39, 7(2013), pp. 1100-1109.

[29] Zhang, J. P.; Chen, F. C.; Li, S. M. Data Stream Clustering Algorithm Based on Density and Affinity Propagation Techniques. // Acta Automatica Sinica, 40, 2(2014), pp. 277-288.

[30] Zhang, X.; Furtlehner, C.; Germain-Renaud, C. et al. Data stream clustering with affinity propagation. // Knowledge and Data Engineering, IEEE Transactions on, 26, 7(2014), pp. 1644-1656. DOI: 10.1109/TKDE.2013.146

Authors' addresses

Limin Wang, Professor
School of Management Science and Information Engineering,
Jilin University of Finance and Economics,
3699 Jingyue Street, Changchun 130117, China
E-mail: wlm_new@163.com

Xing Tao, Postgraduate
School of Management Science and Information Engineering,
Jilin University of Finance and Economics,
3699 Jingyue Street, Changchun 130117, China
E-mail: 459978415@qq.com

Xuming Han, Professor
Corresponding author
School of Computer Science and Engineering,
Changchun University of Technology,
307, Teaching Building, No. 7186, Weixing Road,
Changchun 130012, China
E-mail: hanxvming@163.com

Jialing Han, PhD
School of Computer Science and Engineering,
Changchun University of Technology,
307, Teaching Building, No. 7186, Weixing Road,
Changchun 130012, China
E-mail: weiya2000@163.com

Ying Liu, Associate Professor
School of Computer Science and Engineering,
Changchun University of Technology,
307, Teaching Building, No. 7186, Weixing Road,
Changchun 130012, China
E-mail: lyaihua1995@163.com

Guangyu Mu, Professor
School of Computer Science and Engineering,
Changchun University of Technology,
307, Teaching Building, No. 7186, Weixing Road,
Changchun 130012, China
E-mail: guangyumu@126.com