Chapter 12
Numerical Experiments

12.1 Introduction

This chapter is primarily devoted to the study of the performance of optimization based incremental clustering algorithms. Since the procedure of finding starting cluster centers is an important part of all these algorithms we start the chapter with discussing on the impact of this procedure to the solution obtained by a clustering algorithm.

Then, we demonstrate the performance of the clustering algorithms using data sets with different number of data points and attributes described in Chap. 11: extra small, small, medium sized, large, and very large. The performance profiles are used to evaluate the accuracy of clustering solutions, the number of distance function evaluations and CPU time. In addition, we apply the DB index, the purity, the NMI index and silhouettes to compare different clustering algorithms. In all these algorithms, we consider the MSSC problem. To compare the performance of the incremental clustering algorithms when different similarity measures—d_1, d_2 and d_∞—are used in their objective functions, we apply `DG-Clust` on three real-world data sets given in Chap. 11: German towns, TSPLIB1060 and TSPLIB3038. We use the Voronoi diagrams for this purpose.

12.2 Importance of Procedure for Finding Starting Cluster Centers

In this section, we study the contribution of the procedure for generating starting cluster centers to the quality of the final clustering solutions and also to the overall performance of an incremental clustering algorithm. For this aim, we use three data sets with different number of entries: Ionosphere (small size), Image

A. M. Bagirov et al., *Partitional Clustering via Nonsmooth Optimization*,
Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-030-37826-4_12

segmentation (medium sized) and KEGG metabolic network (large scale). The number of attributes in these data sets ranges from 19 to 34. This choice of data sets allows us to clearly demonstrate the importance of this procedure.

In our experiments, we apply MGKM with and without the procedure for generating starting cluster centers. The performance of this solver with different procedures is compared using three performance measures: accuracy, the number of distance function calculations, and CPU time. The accuracy (or the error) is defined using the formula (10.10).

Consider the k-clustering problem with $k \geq 2$ and recall Algorithm 7.2—the procedure for finding starting cluster centers. This procedure has the following steps:

1: using radii of clusters from the previous iteration determine the list of data points which are candidates to be a starting point;

2: consider each point as the kth cluster center and compute the decrease of the cluster function f_k, defined in (4.4), in comparison with the optimal value f_{k-1}^* for the $(k-1)$-clustering problem;

3: remove some data points from the list using a threshold for the decrease;

4: for each point from the list compute the cluster around it and replace this data point with the corresponding cluster center;

5: solve the auxiliary clustering problem starting from each of these centers.

Steps 1, 2 and 3 use only data points where the preliminary list of candidate starting points is determined. Our aim is to demonstrate that Steps 4 and 5 are very important and make a significant contribution to the quality of the final solution in clustering. Therefore, we consider the following versions of MGKM:

- $V0$: MGKM0—the version where both Steps 4 and 5 are excluded;
- $V1$: MGKM4—the version where only Step 4 is used and Step 5 is excluded;
- $V2$: MGKM5—the version where Step 5 is used and Step 4 is excluded;
- $V3$: MGKM45—the full version with Steps 4 and 5.

The results are presented in Table 12.1, where as before k stands for the number of clusters. In our experiments, we use 5 hours time limit in all versions, that is if the algorithm exceeds this time limit, then its performance is considered as a *failure* (denoted by "fail" in the table). To avoid writing very large numbers for distance function evaluations (denoted by "distance function evals") in the table, we include a number after the name of each data set in brackets and to get the correct values, numbers in distance function evaluations columns should be multiplied by these numbers.

Results show that accuracies of all versions are comparable and differences between them are not significant. Furthermore, as the size of a data set increases the differences become even more insignificant. Regarding the number of distance function evaluations, we can see that the use of the full version $V3$ leads to a significant reduction of the number in all cases. Results for versions $V2$ and $V3$ imply that the use of the auxiliary clustering problem in the procedure allows

Table 12.1 Results with and without the procedure for finding starting points

	Accuracy				Distance function evals				CPU time			
k	V_0	V_1	V_2	V_3	V_0	V_1	V_2	V_3	V_0	V_1	V_2	V_3
Ionosphere ($\times 10^6$)												
2	0.00	0.00	0.00	0.00	1.28	0.55	0.71	0.44	0.33	0.11	0.11	0.06
3	0.03	0.03	0.03	0.03	2.77	1.15	1.24	0.81	0.67	0.22	0.22	0.11
5	0.06	0.07	0.11	0.11	6.28	2.12	2.22	1.51	1.38	0.36	0.36	0.20
7	0.05	0.00	0.10	0.52	10.69	3.40	3.47	2.28	2.14	0.55	0.56	0.30
10	0.27	0.30	0.53	0.28	17.84	5.27	4.91	3.42	3.28	0.81	0.80	0.45
15	0.84	0.65	0.95	2.10	26.07	8.05	6.65	5.05	4.50	1.17	1.06	0.64
20	1.31	1.22	1.54	3.32	34.07	10.34	8.39	6.72	5.70	1.45	1.33	0.86
22	1.50	1.31	1.21	2.80	36.73	11.30	9.05	7.32	6.11	1.59	1.44	0.94
25	1.58	2.00	1.38	3.03	42.81	12.89	9.96	8.42	7.03	1.81	1.56	1.08
Image segmentation ($\times 10^6$)												
2	0.00	0.00	0.00	0.00	0.29	0.22	0.14	0.19	0.27	0.08	0.06	0.05
3	0.00	0.00	0.00	0.00	0.77	0.60	0.36	0.42	0.56	0.22	0.13	0.08
5	0.00	0.00	0.00	0.00	5.38	2.51	2.10	1.28	2.45	0.83	0.48	0.17
7	0.00	2.31	2.30	2.30	12.20	3.92	3.40	2.17	4.41	1.09	0.72	0.30
10	0.00	1.75	1.75	1.75	21.10	7.86	4.82	3.54	6.05	1.81	0.92	0.44
15	0.49	0.49	0.48	1.90	42.58	14.50	8.79	6.61	8.91	2.59	1.45	0.73
20	0.61	0.76	0.79	0.62	92.14	24.05	14.71	11.16	14.94	3.61	2.16	1.14
22	0.64	0.83	0.85	0.66	118.05	29.38	17.93	13.03	17.94	4.16	2.52	1.30
25	0.43	0.86	0.84	0.65	161.32	36.15	23.14	16.95	22.72	4.83	3.14	1.64
KEGG metabolic network ($\times 10^7$)												
2	0.00	0.00	0.00	0.00	0.44	0.35	0.28	0.35	285.28	9.72	10.30	10.28
3	0.00	0.00	0.00	0.00	2.92	1.29	0.79	1.04	4097.22	398.09	201.55	51.69
5	fail	0.07	0.07	0.07	Fail	2.61	1.83	2.19	Fail	794.25	593.31	234.08
7	fail	0.18	0.18	0.18	Fail	3.41	2.55	2.78	Fail	1096.13	842.88	298.98
10	fail	0.01	0.01	0.01	Fail	5.03	4.07	4.10	Fail	1658.86	1311.27	583.70
15	fail	2.96	2.98	2.98	Fail	7.89	6.38	6.48	Fail	1996.03	1514.92	682.95
20	fail	1.26	1.42	1.18	Fail	11.99	10.43	10.48	Fail	2087.19	1677.55	760.66
22	fail	1.74	2.01	1.74	Fail	13.81	12.06	12.33	Fail	2113.00	1698.78	787.09
25	fail	0.21	0.04	1.74	Fail	17.79	15.99	16.10	Fail	2236.72	1936.06	867.17

us to significantly reduce the number of distance function evaluations without deteriorating the final solution. This is due to the fact that the solution obtained by solving the auxiliary clustering problem is close to the solutions of the clustering problem and the k-means algorithm requires a limited number of iterations to obtain them. Similar observations are true also for the required CPU time. Here, we can see that the use of the version $V3$ allows us to significantly decrease the CPU time even in comparison with the versions $V1$ and $V2$.

These results clearly show that regardless of the size of the data sets, the use of the procedure for finding starting cluster centers allows us to significantly reduce

the computational effort while preserving almost the same accuracy for the obtained clustering solutions. Furthermore, the auxiliary clustering problem is an important component of the incremental clustering algorithms.

12.3 Performance Results of Incremental Clustering Algorithms

In this section, we present results on the performance of the incremental clustering algorithms as well as results obtained by the MS-KM. We include the latter algorithm for the comparison purpose. In `MS-KM`, the number of randomly generated starting points is limited by 1000, however, we also applied the time limit which is twice of the CPU time required by `MGKM`. We do not present results of `MS-KM` based on performance profiles using the CPU time and the number of distance function evaluations as the CPU time and also in some sense the number of distance function evaluations are fixed for this algorithm.

We present the results of our experiments for each class of data sets separately. The best known value of the cluster function for a given k is denoted by f_{best}. Note that, in all tables in order to find the true best values of the cluster function, numbers given in the f_{best} column should be multiplied by the number given after the names of data sets.

The error E of a given solution is computed using (10.10). We say that an algorithm finds the best known solutions to the clustering problem if its error $0 \leq E \leq 0.1$. If $0.1 < E \leq 1$, then an algorithm finds nearly the best known solution. For performance profiles, we select in (10.11) and (10.14) the constants $c = \bar{c} = 1$ and the number of iterations to solve a clustering problem $M = 100$.

12.3.1 Results for Extra Small Data Sets

We apply `GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust`, `IS-Clust`, and `MS-KM` to extra small data sets. Other algorithms are not best suited for such data sets. Up to ten clusters are computed in all data sets. Results for accuracy are given in Table 12.2. Note that the best known solutions for all data sets, but Liver disorder data set, are also known to be the global solutions to the corresponding clustering problems.

Results presented in Table 12.2 demonstrate that `MS-KM` cannot be considered as an alternative to any other algorithm. It is able to find the best known solutions only when the number of clusters is small. Otherwise, `MS-KM` fails to find a good quality solution. Other algorithms show the good performance in finding accurate solutions. Nevertheless, most algorithms, except `IS-Clust`, failed to find solutions with high accuracy in Bavaria postal two data set.

Table 12.2 Accuracy results for extra small data sets

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	MS-KM
German towns ($\times 10^5$)								
2	1.21426	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.77009	1.45	0.00	0.00	0.00	0.00	0.00	0.00
4	0.49601	0.72	0.00	0.00	0.00	0.00	0.00	0.00
5	0.38716	0.00	0.00	0.00	0.00	0.00	0.00	0.00
6	0.30535	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.24433	0.09	0.09	0.00	0.09	0.09	0.00	0.00
8	0.21483	1.33	0.69	0.00	0.69	0.69	0.59	3.61
9	0.18669	1.48	1.48	0.00	1.48	1.48	1.37	8.66
10	0.16427	1.06	1.06	0.00	1.06	1.06	0.00	7.70
Bavaria postal 1 ($\times 10^{10}$)								
2	60.25472	7.75	0.00	0.00	0.00	0.00	0.00	7.75
3	29.45066	0.00	0.00	0.00	0.00	0.00	0.00	20.02
4	10.44747	0.00	0.00	0.00	0.00	0.00	0.00	0.08
5	5.97615	0.00	0.00	0.00	0.00	0.00	0.00	23.58
6	3.59085	0.00	28.02	27.79	27.65	28.02	27.65	28.02
7	2.19832	1.50	69.39	0.00	0.00	69.39	69.39	98.03
8	1.33854	0.00	141.13	0.00	0.00	141.13	0.00	225.23
9	0.84237	0.00	259.69	0.00	0.00	259.69	1.44	416.79
10	0.64465	0.00	350.66	0.00	0.00	350.66	0.00	452.52
Bavaria postal 2 ($\times 10^{10}$)								
2	1.99080	162.17	144.28	144.28	144.28	144.28	144.28	144.28
3	1.73988	0.00	0.00	0.00	0.00	0.00	0.00	106.79
4	0.75591	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.53429	1.86	1.14	1.14	1.14	1.14	0.00	1.14
6	0.31876	1.21	39.04	39.04	39.04	39.04	0.00	39.04
7	0.22159	0.50	77.07	76.94	76.94	77.07	0.00	95.00
8	0.17045	0.73	113.22	112.22	112.22	113.21	0.00	153.50
9	0.14011	0.14	142.69	142.48	142.48	142.69	0.00	208.41
10	0.11908	0.16	170.46	169.65	169.65	170.46	0.00	240.56
Iris plant($\times 10^2$)								
2	1.52348	0.00	0.00	0.00	0.00	0.01	0.00	0.00
3	0.78851	0.01	0.00	0.01	0.01	0.01	0.00	0.00
4	0.57228	0.05	0.05	0.00	0.00	0.02	0.00	0.00
5	0.46446	0.54	0.06	0.00	0.00	0.09	0.00	0.06
6	0.39040	1.44	0.07	0.00	0.00	0.10	0.00	0.07
7	0.34298	3.17	0.00	0.00	0.00	0.03	0.00	13.90
8	0.29989	1.71	0.09	0.00	0.00	0.29	0.00	30.27
9	0.27786	2.85	0.10	0.00	0.00	0.40	0.00	40.60
10	0.25834	3.56	0.51	0.00	0.50	0.34	0.06	42.70

(continued)

Table 12.2 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	MS-KM
TSPLIB1060 ($\times 10^{10}$)								
2	0.98319	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.67058	0.00	0.00	0.00	0.00	0.00	0.00	0.00
4	0.47520	0.01	0.01	0.00	0.00	0.01	0.00	0.00
5	0.37910	0.01	0.01	0.01	0.01	0.06	0.00	0.00
6	0.31770	0.06	0.06	0.06	0.06	0.06	0.06	0.00
7	0.27042	0.02	0.02	0.02	0.00	0.02	0.03	0.01
8	0.22643	0.00	0.00	0.00	0.00	0.00	0.02	19.44
9	0.19910	0.30	0.30	0.14	0.00	0.30	0.02	35.81
10	0.17548	0.23	0.04	0.03	0.04	0.23	0.04	28.97
Liver disorders ($\times 10^{6}$)								
2	0.42398	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.32271	0.71	0.71	0.71	0.88	0.88	0.00	0.00
4	0.26066	0.49	0.49	0.11	0.22	0.48	0.22	0.00
5	0.21826	0.08	0.08	0.00	0.07	0.07	0.08	0.01
6	0.18709	0.97	0.05	0.00	0.00	0.14	0.00	0.28
7	0.16420	0.72	0.34	0.00	0.00	0.37	0.00	14.26
8	0.14778	0.41	0.41	0.00	0.00	0.29	0.01	26.95
9	0.13734	0.83	0.00	0.31	0.20	0.31	0.33	36.10
10	0.12742	0.21	0.01	0.00	0.15	0.29	0.00	16.76

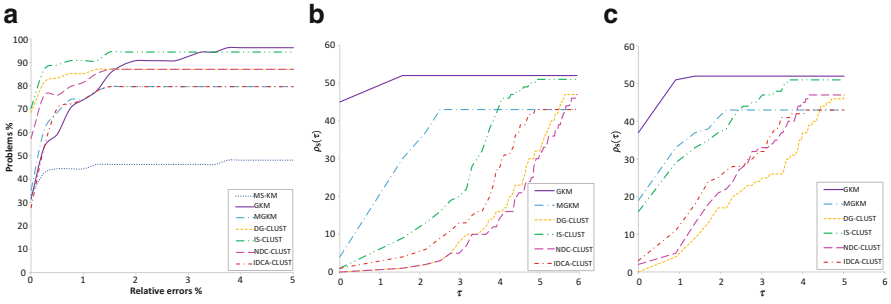

Fig. 12.1 Performance profiles for extra small data sets. (**a**) Relative errors. (**b**) Distance function evals. (**c**) CPU time

Performance profiles for extra small data sets are illustrated in Fig. 12.1. IS-Clust is the most successful in finding the best known solutions and GKM is the most successful in solving clustering problems with the error no more than 5%. MS-KM has the worst performance while GKM requires the least number of distance function evaluations and CPU time. On the other hand, NDC-Clust uses more distance function evaluations and DG-Clust requires more CPU time than any other algorithm.

In Fig. 12.2 graphs for three indices (DB, purity and NMI) and in Fig. 12.3 silhouette plots for $k = 2, 3, 5$ and ten clusters are illustrated using results obtained

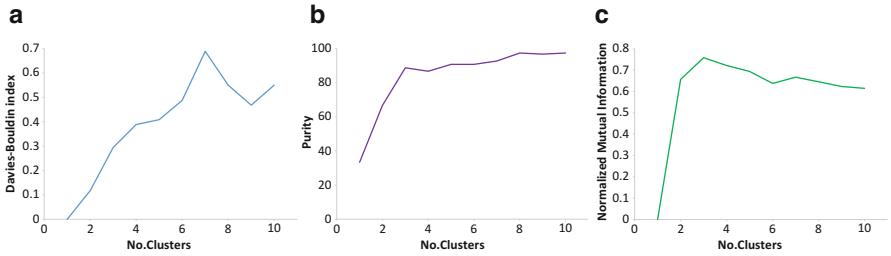

Fig. 12.2 Results for Iris Plant data set using different indices. (**a**) DB index. (**b**) Purity. (**c**) NMI index

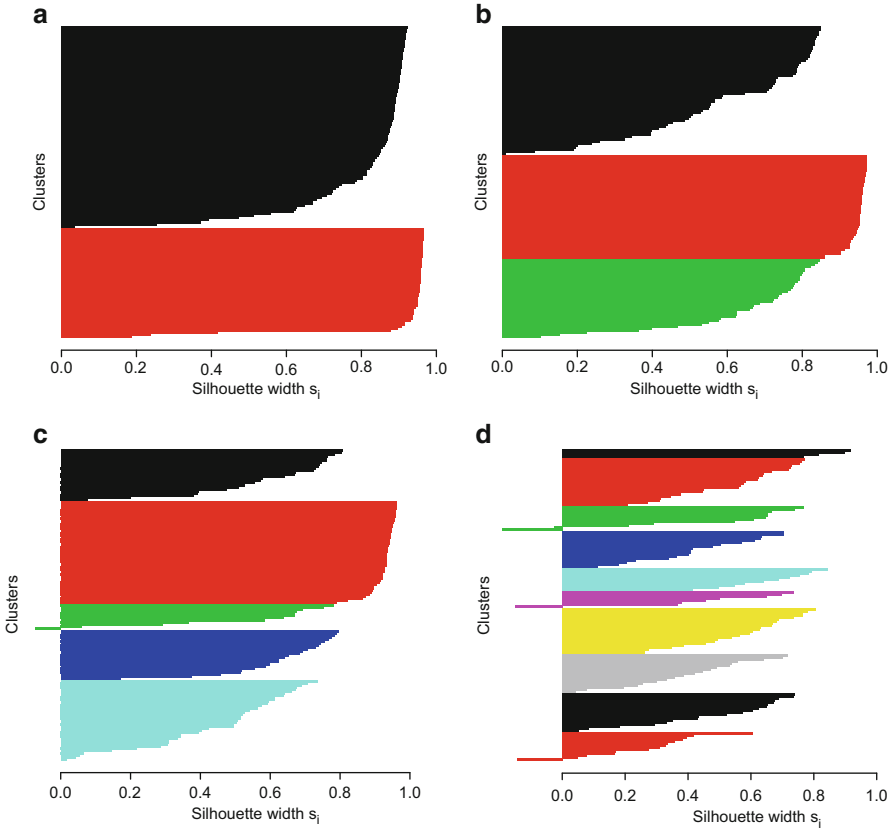

Fig. 12.3 Silhouette plots for Iris plant data set. (**a**) $k = 2$. (**b**) $k = 3$. (**c**) $k = 5$. (**d**) $k = 10$

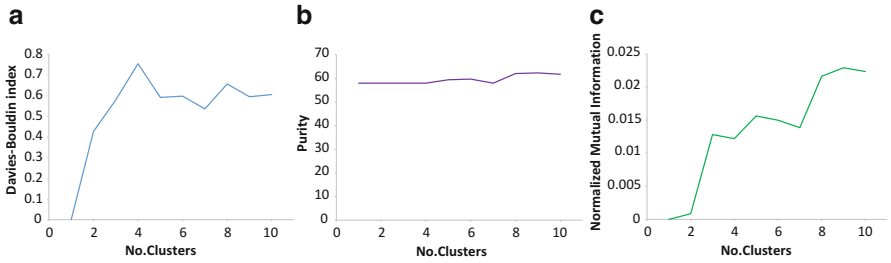

Fig. 12.4 Results for Liver disorder data set using different indices. (**a**) DB index. (**b**) Purity. (**c**) NMI index

by the IS-CLUST in Iris plant data set. The DB index has several knee points at $k = 3, 4, 5, 6$ and one local minimizer at $k = 9$. In general, the purity increases as the number of clusters increases; however, at $k = 3$ it has a local maximizer. The NMI index gets its highest value at $k = 3$. Finally, silhouette plots show that clusters are well-separated when $k = 3$. These results demonstrate a good consistency of the class and the cluster (with $k = 3$) distributions in Iris Plant data set.

In Fig. 12.4 graphs for the three indices and in Fig. 12.5 silhouette plots for $k = 2, 3, 5$ and ten clusters are given based on results obtained by `IS-Clust` in Liver disorder data set. Here, the DB index has three distinct local minimizers at $k = 5, 7$ and $k = 9$. The purity increases as the number of clusters increases; however, this increase is not significant. The NMI index is close to 0 for $k < 3$ and silhouette plots show that clusters are not compact and not well-separated for $k = 2, 3, 5, 10$. Summarizing these results we can conclude that the class and the cluster distributions in Liver disorder data set are incompatible.

12.3.2 Results for Small Data Sets

We apply `GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust`, `IS-Clust`, and `MS-KM` to small data sets. Other algorithms are not well suited for these data sets. Up to 25 clusters are computed in these data sets.

Results for accuracy are given in Table 12.3. The results show that `MS-KM` can reach the best solutions only when the number of clusters is small. Otherwise, this algorithm fails to find high quality solutions. Other algorithms are, in general, successful in finding accurate solutions.

Performance profiles for small data sets are presented in Fig. 12.6. `IS-Clust` is the most successful in finding the best known solutions and in solving clustering problems with the error no more than 5%. As before, `MS-KM` has the worst performance while `GKM` requires the least number of distance function evaluations and CPU time. In addition, `NDC-Clust` uses more distance function evaluations and `DG-Clust` requires more CPU time than any other algorithm.

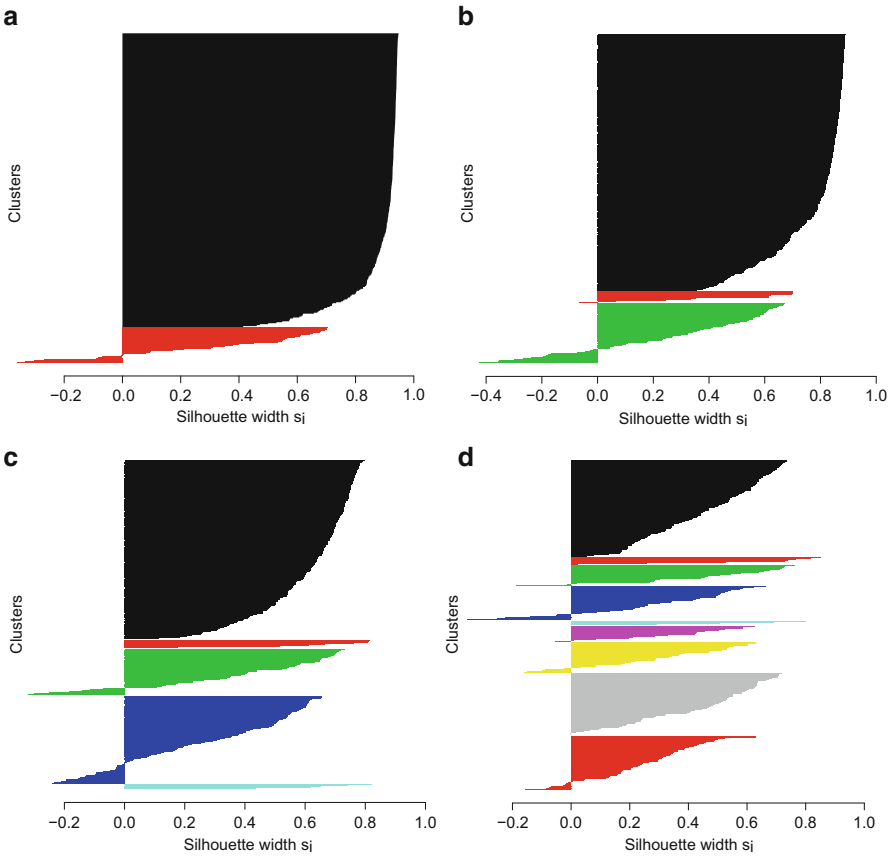

Fig. 12.5 Silhouette plots for Liver disorder data set. (**a**) $k = 2$. (**b**) $k = 3$. (**c**) $k = 5$. (**d**) $k = 10$

In Fig. 12.7, the graph of the DB index is given using the results obtained by `IS-Clust` in TSPLIB3038 data set. Note that the purity and the NMI index require the existence of the class labels and since this data set has no classes we only present the DB index in Fig. 12.7. The silhouette plots for TSPLIB3038 data set with $k = 2, 3, 5$ and ten clusters are given in Fig. 12.8. From Fig. 12.7, it can be observed that the DB index has local minimizers at $k = 5, 7, 11, 15, 20$. Silhouette plots show that in the 10-partition of the data set six clusters are compact and well-separated and other clusters contain some "misclassified" points. The similar observation is true for the k-partitions of the data set with $k = 2, 3$ and 5.

In Fig. 12.9 graphs for the three indices and in Fig. 12.10 silhouette plots for $k = 2, 3, 5, 10$, and 25 clusters are given using results obtained by `IS-Clust` in Vehicle silhouettes data set. The DB index has the distinct local minimizers at $k = 4, 7, 11, 14$ and $k = 19$. The purity increases as the number of clusters increases; however, even for 25 clusters it is only about 55%. The largest value for the NMI

Table 12.3 Accuracy results for small data sets

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	MS-KM
Heart disease ($\times 10^5$)								
2	5.98899	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	4.67508	5.02	3.96	5.14	5.14	5.14	4.67	0.00
5	3.27965	0.52	0.53	0.00	0.01	0.53	0.44	0.03
7	2.64942	2.59	0.00	0.32	0.02	0.00	0.32	4.44
10	2.00558	0.83	0.19	0.00	0.00	0.19	0.03	20.82
15	1.46895	0.55	0.18	0.13	0.18	0.43	0.00	25.37
20	1.16993	0.67	0.00	0.52	0.36	0.51	0.49	37.96
22	1.09199	2.45	0.08	0.40	0.00	0.94	0.23	47.81
25	0.99314	3.33	1.36	0.00	1.13	1.31	0.06	62.52
TSPLIB3038 ($\times 10^9$)								
2	3.16880	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	2.17634	3.43	3.43	3.43	3.43	3.43	3.43	0.00
5	1.19820	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.83967	1.87	1.73	1.85	1.73	1.73	1.73	0.00
10	0.56025	2.78	0.58	0.57	0.58	0.58	0.00	0.00
15	0.35604	0.07	0.05	0.00	0.00	0.07	0.06	0.00
20	0.26681	2.00	0.43	0.14	0.20	0.43	0.16	0.17
22	0.24295	1.64	0.54	0.02	0.00	0.55	0.03	1.19
25	0.21450	0.78	0.43	0.43	0.56	0.43	0.00	1.56
Pima Indians diabetes ($\times 10^6$)								
2	5.14238	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	2.91332	1.23	1.23	1.23	1.23	1.23	1.23	0.00
5	1.73687	0.15	0.15	0.00	0.15	0.15	0.01	0.01
7	1.30315	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.93066	1.84	0.06	0.00	1.66	0.06	1.12	12.66
15	0.69579	0.21	0.00	1.06	0.23	0.05	1.37	34.14
20	0.57278	0.28	0.00	0.18	0.10	0.35	0.27	47.39
22	0.53501	0.55	0.34	0.33	0.00	0.38	0.14	55.16
25	0.48874	0.38	0.38	0.35	0.00	0.43	0.17	67.21
Breast cancer Wisconsin ($\times 10^4$)								
2	1.93232	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	1.62555	0.01	0.01	0.00	0.00	0.01	0.00	0.00
5	1.37047	2.28	0.01	0.02	0.00	0.02	0.00	0.00
7	1.20497	1.44	0.12	0.00	0.02	0.10	0.00	6.27
10	1.01996	0.16	0.32	0.04	0.13	0.27	0.00	17.41
15	0.86928	1.02	0.58	0.55	0.68	0.97	0.00	23.39
20	0.76651	3.40	0.69	0.68	0.53	1.42	0.00	31.34
22	0.72906	5.37	2.12	0.48	1.35	2.42	0.00	32.59
25	0.69446	4.48	0.35	0.00	1.12	2.41	0.38	32.68

(continued)

Table 12.3 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	MS-KM
Ionosphere ($\times 10^4$)								
2	0.24194	0.00	0.00	0.00	0.00	0.00	0.00	2.75
3	0.21933	0.96	0.03	0.89	0.02	0.03	0.00	2.45
5	0.18908	0.11	0.11	0.00	0.10	0.11	0.13	2.20
7	0.17382	0.46	0.53	0.00	0.38	0.53	0.03	3.28
10	0.15540	2.74	0.27	0.00	0.22	0.32	0.12	5.11
15	0.13729	6.48	2.10	0.92	1.61	1.43	0.00	6.47
20	0.12307	9.23	3.32	2.09	2.79	2.73	0.00	13.52
22	0.11839	9.70	2.80	1.62	2.00	2.92	0.00	15.27
25	0.11147	10.00	3.03	1.38	1.86	2.98	0.00	21.06
Vehicle silhouettes ($\times 10^6$)								
2	7.29088	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	4.87412	0.02	0.02	0.00	0.00	0.02	0.02	0.00
5	2.36484	0.00	0.00	0.00	0.04	0.00	0.00	0.00
7	1.71738	1.08	0.00	0.00	0.00	0.00	0.00	1.72
10	1.25217	0.68	0.04	0.52	0.01	0.22	0.00	21.48
15	0.89095	1.03	0.02	0.00	0.07	0.08	0.00	24.49
20	0.74221	1.17	0.26	0.09	0.00	0.28	0.12	15.44
22	0.69630	0.68	0.09	0.00	0.02	0.09	0.03	20.05
25	0.63106	1.10	0.07	0.00	0.03	0.08	0.01	31.18

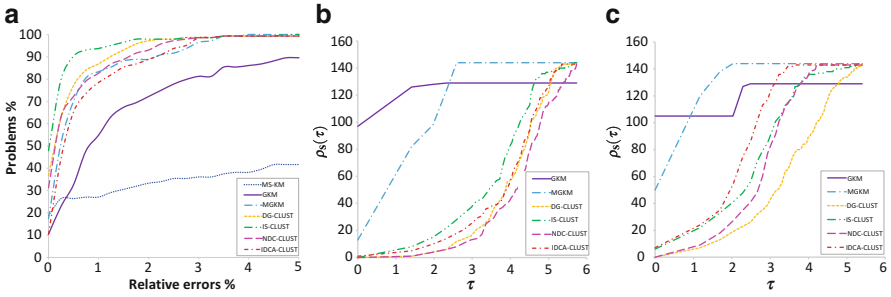

Fig. 12.6 Performance profiles for small data sets. (**a**) Relative errors. (**b**) Distance function evals. (**c**) CPU time

index is about 0.22 for $k = 5$ and $k = 7$. Silhouette plots show that not all clusters are well-separated in k-partitions with $k = 2, 3, 5, 10$ and 25. For instance, five clusters are not well-separated when $k = 25$. These results demonstrate that in vehicle silhouettes data set the class and the cluster distributions are not consistent.

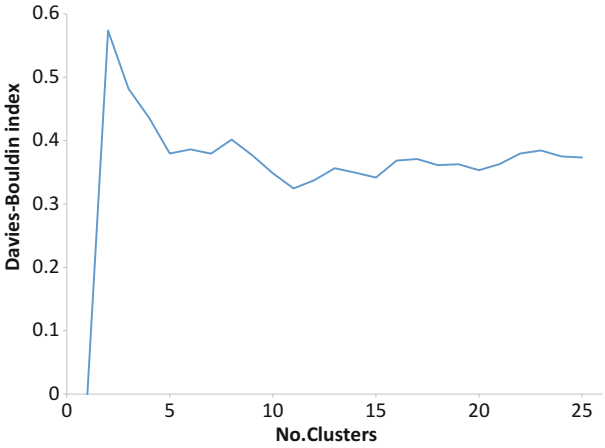

Fig. 12.7 DB index for TSPLIB3038 data set

12.3.3 Results for Medium Sized Data Sets

`GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust`, `IS-Clust`, `LMB-Clust` and `DCDB-Clust` are applied to medium sized data sets. Results for accuracy are given in Table 12.4. These results demonstrate that, overall, all algorithms, except `DCDB-Clust`, are able to find the best known solutions.

Performance profiles for medium sized data sets are depicted in Fig. 12.11. They show that `IS-Clust` is the most successful algorithm in finding the best known solutions and `GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust` and `IS-Clust` are all successful in solving clustering problems with the error no more than 5%. `DCDB-Clust` has the worst performance both in terms of errors and distance function calls. `MGKM` requires the least number of distance function evaluations whereas `LMB-Clust` requires the least CPU time among all algorithms. Finally, `GKM` uses more CPU time than other algorithms.

In Fig. 12.12 graphs of the three indices and in Fig. 12.13 silhouette plots for $k = 2, 3, 5, 10$ and 25 clusters are illustrated using results obtained by `IS-Clust` in Image segmentation data set. The DB index has local minimizers at $k = 3, 7, 16, 21$ and $k = 3$ is a global minimizer. Overall, the purity shows the steady increase as the number of clusters increases; however, it becomes almost a constant after $k = 16$. The NMI index has the large value at $k = 7$ and the largest value at $k = 14$. Note that the number of classes in this data set is 7. Results for silhouettes demonstrate that a large portion of clusters are not well-separated. Summarizing, we can say that in this data set there is some compatibility between the class and the cluster distributions but it is not very high.

The graph of the DB index for Pla85900 data set is depicted in Fig. 12.14. The DB index has local minimizers at $k = 4, 8, 11, 13$ and $k = 21$. Among them $k = 4$,

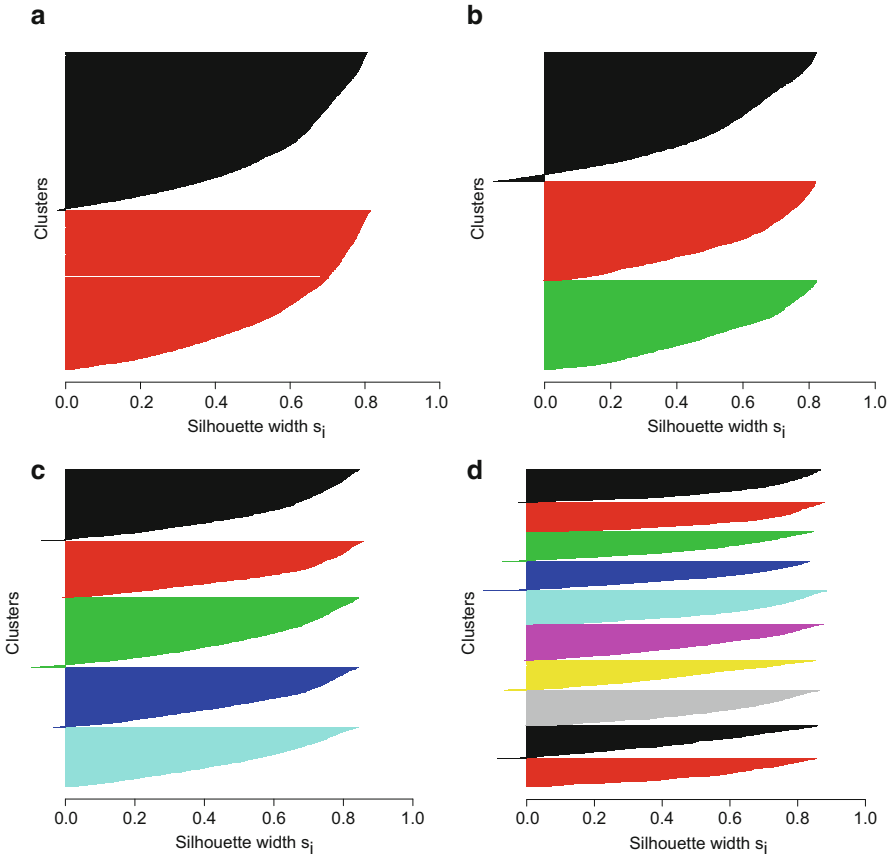

Fig. 12.8 Silhouette plots for TSPLIB3038 data set. (**a**) $k = 2$. (**b**) $k = 3$. (**c**) $k = 5$. (**d**) $k = 10$

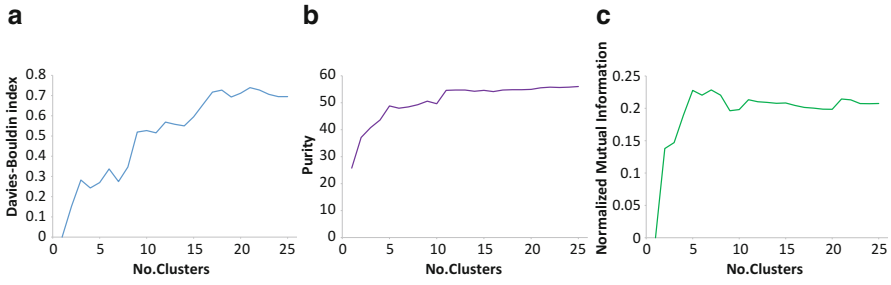

Fig. 12.9 Results for Vehicle silhouettes data set using different indices. (**a**) DB index. (**b**) Purity. (**c**) NMI index

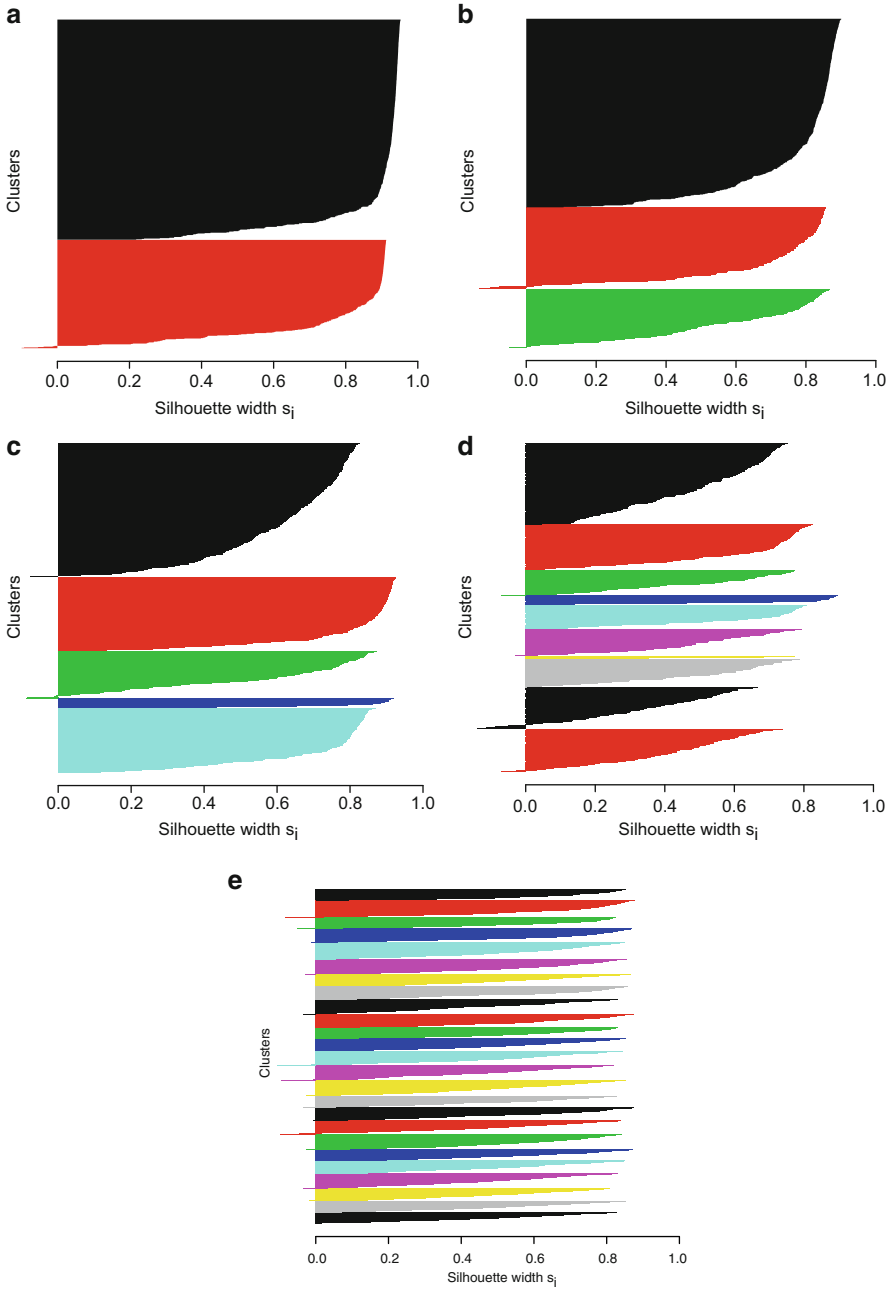

Fig. 12.10 Silhouette plots for Vehicle silhouettes data set. (**a**) $k = 2$. (**b**) $k = 3$. (**c**) $k = 5$. (**d**) $k = 10$. (**e**) $k = 25$

Table 12.4 Accuracy results for medium sized data sets

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
DC15112 ($\times 10^{11}$)									
2	3.68403	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	2.53240	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	1.32707	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.93208	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.64491	1.41	1.41	1.41	1.41	1.41	1.41	1.41	1.41
15	0.43136	0.26	0.26	0.00	0.24	0.25	0.24	0.23	0.00
20	0.32178	0.24	0.24	0.24	0.24	0.24	0.24	0.24	0.00
22	0.28995	0.73	0.20	0.20	0.00	0.20	0.00	0.00	0.00
25	0.25428	0.01	0.01	0.01	0.00	0.01	0.18	0.00	0.01
Image segmentation ($\times 10^{7}$)									
2	3.56057	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
3	2.74163	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.76
5	1.71429	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	1.34043	2.30	2.30	2.30	2.30	2.31	1.79	2.62	5.26
10	0.97955	1.75	1.75	1.75	1.75	1.75	1.15	1.75	6.50
15	0.65554	0.10	1.91	1.89	1.89	1.90	1.72	1.90	11.98
20	0.51300	0.06	0.63	0.65	0.65	0.64	0.02	0.00	15.66
22	0.47112	0.06	0.67	0.71	0.67	0.70	0.00	0.39	23.14
25	0.41605	0.11	0.82	0.80	0.80	0.82	0.05	0.00	9.54
Page blocks ($\times 10^{10}$)									
2	5.79368	0.24	0.00	0.00	0.00	0.00	0.00	0.24	0.00
3	3.31337	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00
5	1.32184	0.00	0.00	0.00	0.03	0.00	0.19	0.00	0.25
7	0.82934	0.18	0.18	0.18	0.16	0.18	0.00	0.03	0.19

(continued)

Table 12.4 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
10	0.45330	1.53	1.53	1.51	1.54	1.53	1.51	1.47	0.68
15	0.24936	1.02	1.87	1.65	1.72	1.65	1.84	4.57	4.84
20	0.17139	0.00	1.54	2.11	2.11	2.11	1.54	4.47	3.73
22	0.14988	0.75	0.79	0.93	0.93	0.79	0.80	3.58	0.00
25	0.12034	2.64	2.01	2.08	2.09	2.01	2.02	4.89	0.00
Pla85900 ($\times 10^{15}$)									
2	3.74908	0.00	0.00	0.00	0.00	0.00	0.00	1.44	0.00
3	2.28057	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	1.33972	0.00	0.00	0.00	0.00	0.00	0.00	2.77	0.00
7	0.97110	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.68294	0.00	1.03	0.00	0.00	0.00	0.00	0.40	0.00
15	0.46029	0.51	0.98	0.51	0.51	0.98	0.02	0.92	0.48
20	0.34986	0.29	0.29	0.52	0.52	0.52	0.52	0.95	0.52
22	0.31942	0.10	0.09	0.00	0.19	0.19	0.17	0.47	0.46
25	0.28223	1.09	0.14	0.13	0.30	0.30	0.00	0.30	0.30
Pen-based recognition of handwritten digits ($\times 10^{8}$)									
2	1.28119	0.39	0.00	0.00	0.00	0.00	0.00	0.00	0.39
3	1.01594	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.75304	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.17
7	0.59993	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.49302	0.00	0.00	0.00	0.00	0.00	0.00	2.63	0.00
15	0.39067	0.00	0.00	0.00	0.00	0.00	0.00	1.75	0.00
20	0.34123	0.00	0.17	0.16	0.17	0.17	0.17	0.94	0.17
22	0.32312	0.00	0.01	0.00	0.00	0.00	0.00	1.17	0.55
25	0.30109	0.00	0.00	0.00	0.00	0.00	0.00	0.87	0.00

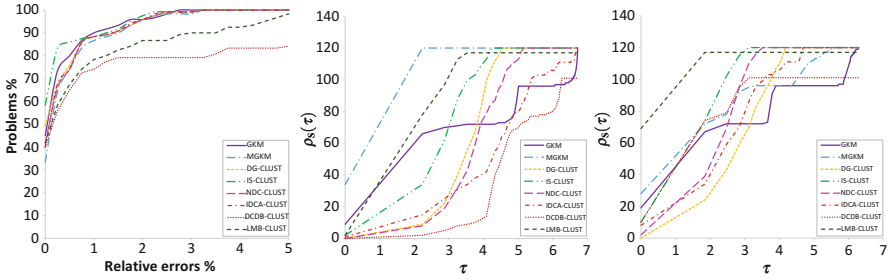

Fig. 12.11 Performance profiles for medium sized data sets. (**a**) Relative errors. (**b**) Distance function evals. (**c**) CPU time

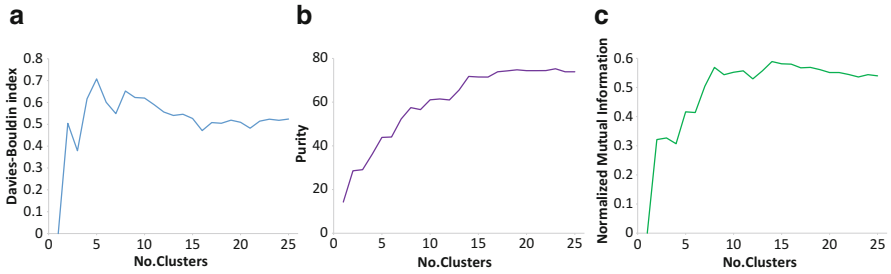

Fig. 12.12 Results for Image segmentation data set using different indices. (**a**) DB index. (**b**) Purity. (**c**) NMI index

$k = 8$ and $k = 21$ are global minimizers or nearly global minimizers. The deep global minimizer is located at $k = 4$.

12.3.4 Results for Large Data Sets

We apply `GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust`, `IS-Clust`, `LMB-Clust`, and `DCDB-Clust` to large data sets. Results for accuracy are given in Tables 12.5 and 12.6. Note that in these tables the values of f_{best} are the best values obtained by algorithms used in our numerical experiments. Results demonstrate that all algorithms are successful in finding best known solutions.

Performance profiles for large data sets are presented in Fig. 12.15. As before, `IS-Clust` is the most successful in finding the best solutions, and `DG-Clust`, `IS-Clust`, `NDC-Clust`, `IDCA-Clust` are successful in solving clustering problems with the error no more than 5%. `LMB-Clust` requires the least and `GKM` the largest number of distance function evaluations. In addition, `LMB-Clust` requires the least CPU time among all algorithms. `GKM` uses more CPU time than other algorithms.

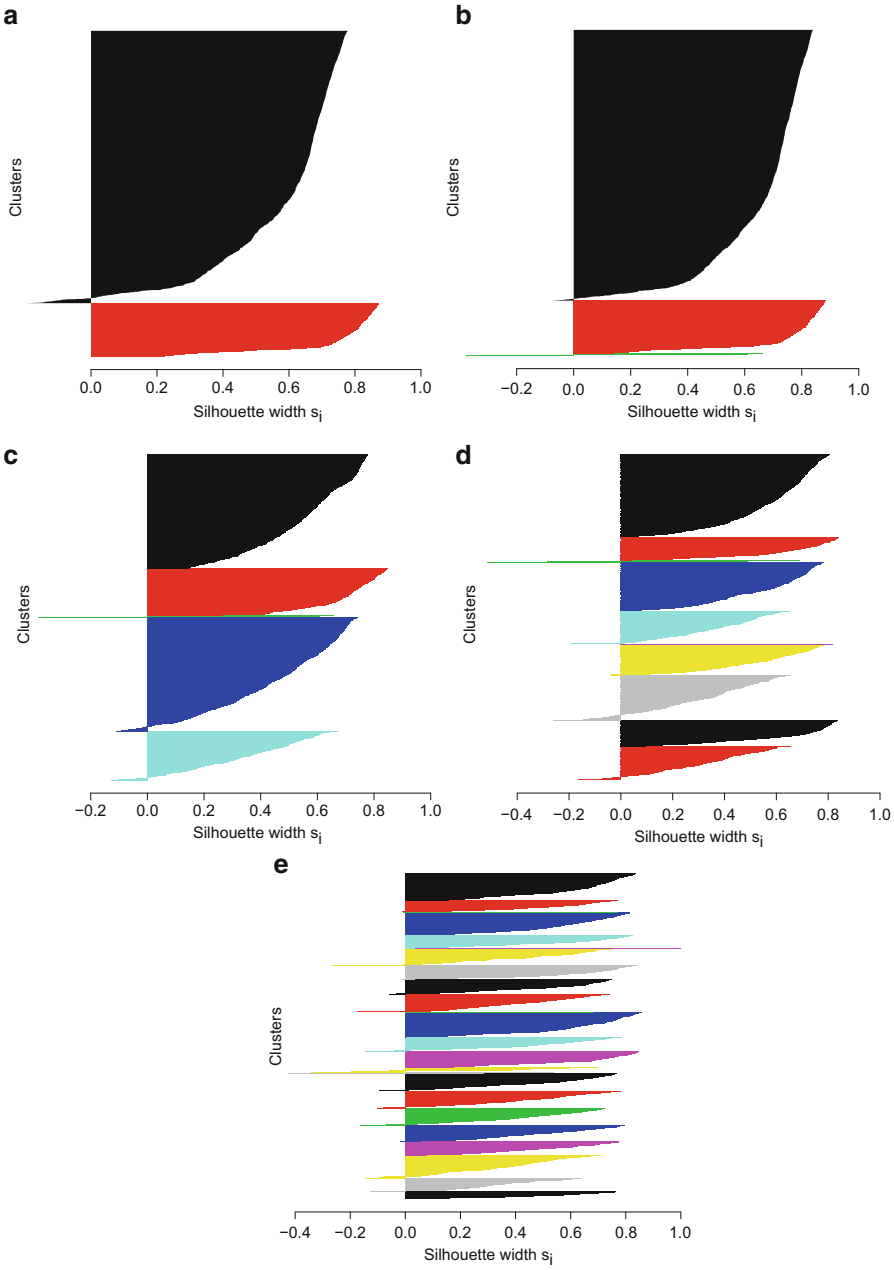

Fig. 12.13 Silhouette plots for Image segmentation data set. (**a**) $k = 2$. (**b**) $k = 3$. (**c**) $k = 5$. (**d**) $k = 10$. (**e**) $k = 25$

Table 12.5 Accuracy results for large data sets

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
Landsat satellite ($\times 10^7$)									
2	5.12686	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
3	2.50603	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01
5	1.82661	0.00	1.70	1.69	1.70	1.70	1.69	1.70	1.70
7	1.50121	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02
10	1.23428	0.00	2.03	2.02	2.02	2.02	2.02	1.74	2.06
15	1.02265	0.06	0.10	0.07	0.00	0.08	0.88	0.88	0.00
20	0.91254	0.90	0.02	0.00	0.82	0.01	0.82	1.93	0.90
22	0.87952	0.11	0.27	0.24	0.22	0.26	0.00	0.26	1.25
25	0.83690	0.10	0.05	0.03	0.06	0.07	0.00	0.66	0.74
Optical recognition of handwritten digits ($\times 10^7$)									
2	0.60042	0.00	0.00	0.00	0.00	0.00	0.00	0.63	0.01
3	0.54582	0.00	0.00	0.00	0.00	0.00	0.00	0.75	0.01
5	0.47140	0.56	1.01	1.00	0.00	1.01	0.00	0.82	0.56
7	0.41601	0.00	0.00	0.00	0.88	0.00	0.89	0.86	0.08
10	0.36777	0.00	0.80	0.79	0.60	0.79	0.79	0.60	0.00
15	0.32522	0.92	0.90	0.90	0.90	0.90	0.00	1.86	0.90
20	0.30030	0.78	0.01	0.00	0.00	0.00	0.00	1.04	0.32
22	0.29376	0.00	0.18	0.16	0.16	0.17	0.16	0.19	0.17
25	0.28427	0.23	0.07	0.03	0.03	0.07	0.02	0.19	0.00

(continued)

Table 12.5 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
Letters recognition ($\times 10^6$)									
2	1.38189	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04
3	1.25058	0.00	0.00	0.00	0.00	0.00	0.00	4.19	0.20
5	1.08652	1.06	0.00	0.00	0.00	0.00	0.00	0.00	1.06
7	0.97240	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.49
10	0.85750	0.00	0.19	0.18	0.19	0.19	0.00	1.10	0.21
15	0.74761	0.08	0.09	0.39	0.07	0.09	0.00	0.30	0.00
20	0.67601	0.22	0.03	0.00	0.03	0.04	0.02	1.13	0.31
22	0.65355	1.02	0.84	0.00	0.48	0.84	0.00	0.84	0.42
25	0.62338	1.39	0.41	0.00	0.36	0.41	0.41	0.65	0.89
EEG eye state ($\times 10^8$)									
2	8178.14	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	1833.88	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	1.33858	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	0.67714	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.01
10	0.45669	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
15	0.34653	0.00	1.09	0.69	0.69	1.09	0.93	0.05	0.28
20	0.28985	0.01	1.32	0.09	0.96	1.32	0.00	0.00	1.54
22	0.27622	0.28	0.78	1.02	0.69	0.75	0.00	0.29	0.81
25	0.25979	0.67	0.00	0.24	0.31	0.75	0.23	0.20	0.08

Shuttle control ($\times 10^8$)									
2	21.34329	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	10.85415	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	7.24479	0.00	0.00	0.00	0.07	0.01	0.00	0.09	5.39
7	4.33840	0.01	0.00	0.00	0.52	0.00	0.00	0.37	0.65
10	2.83166	0.02	0.00	0.00	0.59	0.01	0.00	0.57	0.85
15	1.53154	0.00	0.00	0.00	3.16	0.02	0.00	0.02	7.48
20	1.06012	0.00	0.00	0.15	1.32	0.02	0.00	0.03	4.33
22	0.94081	0.16	0.08	0.16	2.49	0.02	0.00	0.03	5.85
25	0.77978	2.48	2.48	2.47	2.05	2.52	2.48	0.00	10.12
Person activity ($\times 10^5$)									
2	1.03715	0.00	0.00	0.00	0.00	0.02	0.00	0.00	1.50
3	0.77228	0.00	0.00	0.00	0.00	0.03	0.00	0.00	4.29
5	0.56018	0.00	0.00	0.00	0.00	0.10	0.00	0.00	2.30
7	0.44514	0.00	0.00	0.00	0.00	0.15	0.00	1.20	1.10
10	0.33412	0.00	7.02	5.46	7.02	7.33	7.57	7.02	0.00
15	0.26151	0.54	1.09	0.00	0.00	2.05	0.00	0.00	0.00
20	0.21791	1.00	1.00	0.83	0.83	1.83	0.83	0.83	0.00
22	0.20573	0.01	1.22	1.32	1.32	1.38	0.00	1.15	0.70
25	0.18890	0.00	0.00	0.00	0.00	1.80	0.00	2.84	0.71

(continued)

Table 12.5 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
KEGG metabolic relation network ($\times 10^8$)									
2	11.38530	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	4.90060	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	1.88367	0.07	0.00	0.00	0.00	0.00	0.00	0.00	0.07
7	1.19630	0.19	0.03	0.08	0.82	0.06	0.00	2.20	1.69
10	0.63515	0.01	0.00	0.01	0.32	0.00	0.00	0.00	1.01
15	0.35122	3.41	4.34	1.04	1.04	1.03	0.00	1.48	2.21
20	0.24984	1.21	1.78	2.12	1.34	2.08	0.00	3.50	0.40
22	0.22365	1.00	0.00	1.00	0.29	1.00	2.03	4.95	3.53
25	0.19289	0.53	0.51	0.01	1.53	0.00	1.68	1.64	4.02
Skin segmentation ($\times 10^9$)									
2	1.32236	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.89362	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.50203	0.00	0.00	0.00	0.00	0.00	1.65	0.00	0.00
7	0.36308	0.00	0.00	0.00	0.00	0.00	0.00	11.43	0.00
10	0.25122	0.01	4.25	4.25	4.25	4.25	4.25	13.37	0.00
15	0.16963	0.02	0.19	0.00	0.00	0.00	0.05	3.84	0.00
20	0.12615	Fail	0.00	1.70	1.70	1.71	0.21	4.50	1.20
22	0.11654	Fail	0.57	0.73	0.73	0.75	0.00	6.36	0.75
25	0.10228	Fail	0.00	0.70	0.70	0.70	0.00	5.79	0.70

3D Road network ($\times 10^6$)									
2	49.13298	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	22.77818	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03
5	8.82574	Fail	0.00	0.00	0.00	0.01	0.00	0.00	0.00
7	4.84744	Fail	0.00	0.00	0.01	0.04	0.00	0.00	0.94
10	2.56662	Fail	Fail	0.01	0.02	0.23	0.00	0.00	0.02
15	1.27069	Fail	Fail	0.00	0.01	0.62	0.00	0.00	0.98
20	0.80865	Fail	Fail	0.00	0.01	0.48	0.00	0.00	0.03
22	0.70328	Fail	Fail	0.00	0.00	0.42	0.00	0.00	4.14
25	0.59258	Fail	Fail	1.94	3.79	2.39	1.94	0.00	0.30
Gas sensor array drift ($\times 10^{13}$)									
2	7.91182	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	5.02409	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	3.22395	0.10	0.10	0.10	0.10	0.10	0.10	0.00	0.10
7	2.25010	0.00	0.05	0.00	0.00	0.00	0.00	0.00	0.00
10	1.65228	0.18	0.18	0.18	0.18	0.18	0.00	0.18	0.18
15	1.14212	0.00	0.00	0.00	0.00	0.00	0.15	0.00	1.25
20	0.87878	0.67	1.58	0.01	0.01	1.58	0.00	2.74	1.54
22	0.80882	1.12	3.22	0.00	0.00	3.21	0.27	3.49	1.37
25	0.72211	0.65	2.58	0.16	0.16	2.53	0.00	3.05	0.66

Table 12.6 Accuracy results for very large data sets

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
Isolet ($\times 10^6$)									
2	0.72194	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.67878	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.69
5	0.61365	2.44	1.04	0.39	1.04	1.04	1.04	0.00	0.00
7	0.57029	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.22
10	0.53478	1.64	0.12	0.12	0.12	0.12	0.14	1.64	0.00
15	0.48739	0.10	0.00	0.00	0.00	0.00	0.55	0.54	0.10
20	0.46045	0.04	0.03	0.00	0.00	0.03	0.17	0.12	0.16
22	0.45459	0.00	0.60	0.04	0.04	0.57	0.74	0.09	0.11
25	0.44389	0.23	0.18	0.15	0.14	0.15	3.16	0.40	0.00
Online news popularity ($\times 10^{14}$)									
2	9.53913	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	5.91077	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	3.09885	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	1.79526	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	1.17247	0.00	0.00	0.02	0.00	0.00	0.00	2.57	0.00
15	0.77637	0.00	0.00	0.00	0.00	0.00	0.00	14.77	0.00
20	0.59812	0.02	0.11	0.12	0.10	0.11	0.00	7.40	0.11
22	0.55266	0.61	2.24	2.24	0.11	0.11	0.00	5.26	0.69
25	0.49615	0.13	0.26	0.25	0.13	0.13	0.00	7.01	0.99

Sensorless drive diagnosis ($\times 10^7$)									
2	3.88116	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	2.91313	0.00	0.00	0.00	0.02	0.17	0.00	0.00	8.31
5	1.93652	0.01	0.01	0.00	0.03	0.23	0.00	5.32	10.67
7	1.53488	0.01	0.01	0.00	0.03	0.27	0.00	0.00	4.18
10	0.96091	0.00	0.00	0.14	3.82	0.44	0.00	4.41	12.62
15	0.62816	0.00	0.00	0.11	2.83	0.46	0.00	0.00	3.69
20	0.49989	0.00	3.92	6.83	5.63	4.25	3.89	0.80	5.29
22	0.46915	3.36	3.53	13.83	4.53	3.57	3.30	0.00	4.93
25	0.42232	14.82	6.22	26.45	7.16	6.42	6.18	0.00	2.85
Covertype ($\times 10^{11}$)									
2	1.34188	Fail	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	0.95287	Fail	0.00	0.00	0.00	0.00	0.00	0.00	0.00
5	0.58977	Fail	Fail	0.00	0.00	0.00	0.00	0.00	0.00
7	0.44828	Fail	Fail	0.00	0.00	0.00	0.00	0.00	0.00
10	0.33878	Fail	Fail	0.00	0.00	0.00	0.00	0.00	0.00
15	0.24669	Fail	Fail	0.00	0.00	0.87	0.87	0.00	0.87
20	0.20395	Fail	Fail	0.00	0.00	0.00	0.00	0.05	Fail
22	0.18951	Fail	Fail	0.00	0.00	3.58	0.00	1.55	Fail
25	0.17362	Fail	Fail	0.12	0.12	13.06	0.12	0.00	Fail

(continued)

Table 12.6 (continued)

k	f_{best}	GKM	MGKM	DG-Clust	NDC-Clust	IDCA-Clust	IS-Clust	LMB-Clust	DCDB-Clust
MiniBooNE particle identification ($\times 10^{10}$)									
2	8.92236	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3	5.22601	0.00	0.00	0.00	0.00	0.00	0.00	21.68	0.00
5	1.82252	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00
7	1.29369	Fail	0.00	0.00	0.00	0.00	0.00	0.00	0.00
10	0.92406	Fail	0.01	0.01	2.85	2.85	5.50	0.00	0.03
15	0.63507	Fail	0.00	0.00	0.01	0.00	0.00	0.00	0.01
20	0.50871	Fail	0.35	0.38	0.38	Fail	0.35	1.15	0.00
22	0.47966	Fail	1.48	Fail	0.23	Fail	Fail	1.49	0.00
25	0.44425	Fail	0.01	Fail	0.04	Fail	Fail	0.00	0.02
Gisette ($\times 10^{13}$)									
2	0.41994	Fail	0.00	Fail	0.00	Fail	Fail	0.00	0.00
3	0.41160	Fail	0.00	Fail	0.00	Fail	Fail	0.00	0.69
5	0.40232	Fail	0.00	Fail	0.00	Fail	Fail	0.00	1.34
7	0.39535	Fail	1.76	Fail	0.01	Fail	Fail	0.00	2.45
10	0.38843	Fail	Fail	Fail	0.00	Fail	Fail	0.00	2.95
15	0.38177	Fail	Fail	Fail	0.34	Fail	Fail	0.00	Fail
20	0.38144	Fail	Fail	Fail	0.43	Fail	Fail	0.00	Fail
22	0.37843	Fail	Fail	Fail	1.23	Fail	Fail	0.00	Fail
25	0.37344	Fail	Fail	Fail	Fail	Fail	Fail	0.40	Fail

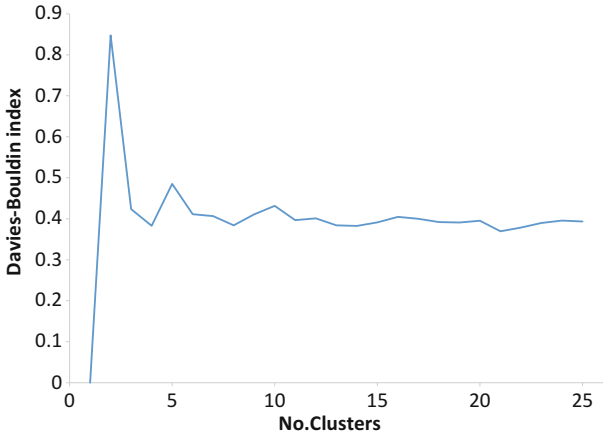

Fig. 12.14 DB index for Pla85900 data set

In Fig. 12.16 graphs of the three indices are depicted using results obtained by `IS-Clust` in Gas sensor array drift data set. It can be seen that the DB index has local minimizers at $k = 4, 6, 11, 17, 24$ and $k = 4$ is the global minimizer. The purity increases up to about 55% as the number of clusters increases. The NMI index has the largest values at $k = 17$ and $k = 22, 23, 24, 25$ with the value 0.34. Note that the number of classes in this data set is 6. The results show that in this data set the level of the compatibility between the class and the cluster distributions is not high.

The graph of the DB index for KEGG metabolic relation network data set is presented in Fig. 12.17. Here, the DB index has many local minimizers. Two of them are global minimizers ($k = 7$ and $k = 10$). This means that the most compact and well-separated clusters for this data set obtained for the 7- and 10-partitions.

12.3.5 Results for Very Large Data Sets

`GKM`, `MGKM`, `DG-Clust`, `NDC-Clust`, `IDCA-Clust`, `IS-Clust`, `LMB-Clust` and `DCDB-Clust` are applied to very large data sets. Results for accuracy are given in Table 12.6. Note that in these tables the values of f_{best} are the best values obtained by all algorithms used in the numerical experiments.

We can see that not all algorithms are able to solve clustering problems within the given 5 h time limit. `GKM` fails in three largest data sets, `MGKM` fails in two of them, `IDCA-Clust` and `IS-Clust` fail in one of them. This means that these algorithms are not always applicable to solve clustering problems in very large data sets. However, `LMB-Clust` succeeds to solve all problems within the given 5 h time limit.

Performance profiles for very large data sets are presented in Fig. 12.18. It can be observed that `LMB-Clust` is the most successful in finding the best known

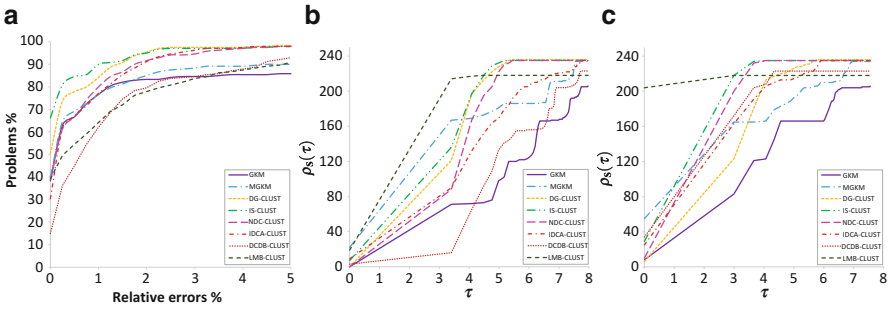

Fig. 12.15 Performance profiles for large data sets. (**a**) Relative errors. (**b**) Distance function evals. (**c**) CPU time

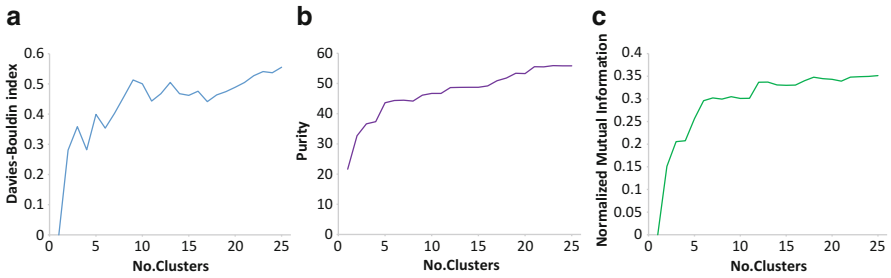

Fig. 12.16 Results for Gas sensor array drift data set using different indices. (**a**) DB index. (**b**) Purity. (**c**) NMI index

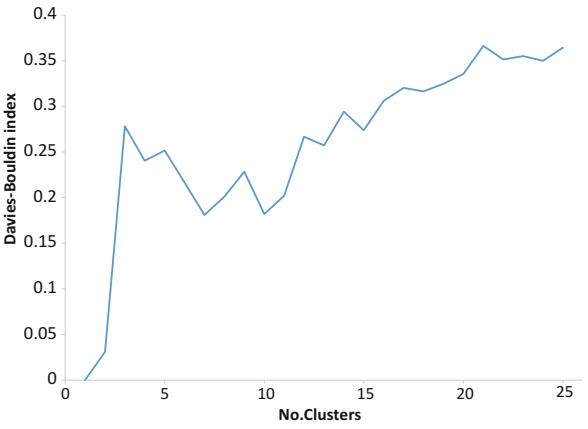

Fig. 12.17 DB index for KEGG metabolic relation network data set

solutions and NDC-Clust is the most successful in solving clustering problems with the error no more than 5%. In addition, LMB-Clust has the least number of distance function evaluations and CPU time. The results also show that GKM is not applicable to very large data sets, it requires largest number of distance function evaluations and CPU time among all algorithms.

In Fig. 12.19 graphs of the DB index and purity are presented based on results obtained by IS-Clust in Covertype data set. It can be seen from the figure that the DB index has local minimizers at $k = 3, 8, 21$ and $k = 8$ is a global minimizer. The DB index tends to increase as the number of clusters increases. This can be considered as an indication that according to the DB index the 8-partition of Covertype data set has the best separated clusters among all k-partitions ($k = 2, \ldots, 25$).

The purity tends to increase starting from 48% up to about 52% as the number of clusters increases from 2 to 25. This means that we should calculate a large number of clusters to get a significant increase of the purity in Covertype data set.

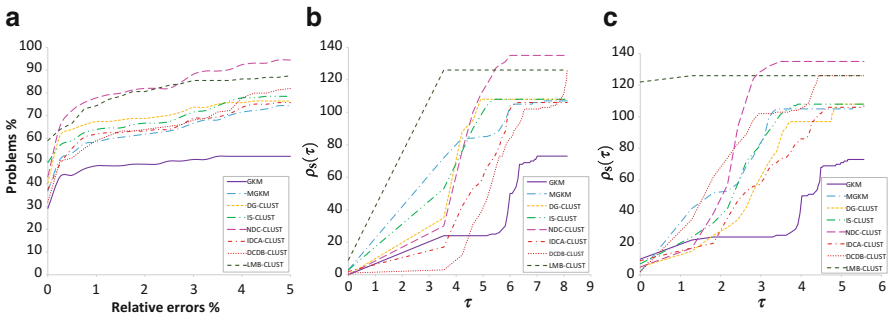

Fig. 12.18 Performance profiles for very large data sets. (**a**) Relative errors. (**b**) Distance function evals. (**c**) CPU time

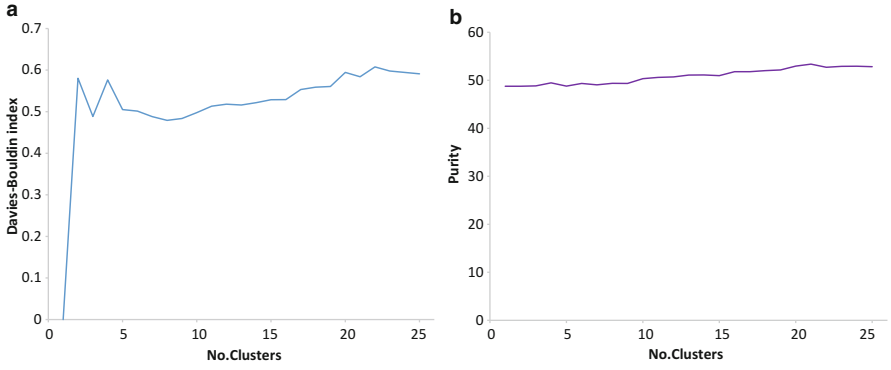

Fig. 12.19 DB index and purity for Covertype data set. (**a**) DB index. (**b**) Purity

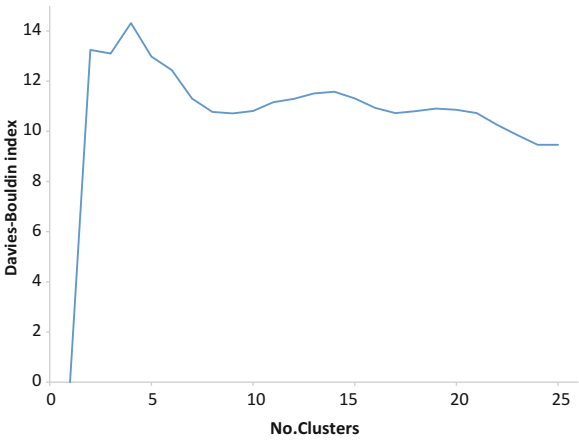

Fig. 12.20 DB index for Gisette data set

The graph of the DB index for Gisette data set is presented in Fig. 12.20. This index has three distinct local minimizers at $k = 8$, 16 and $k = 24$. However, it tends to decrease as the number of clusters increases. This means that we need to compute a large number of clusters to find a cluster distribution with well-separated clusters. This is not unexpected for Gisette data set as it has 5000 attributes and is sparse.

12.4 Comparative Results with Different Similarity Measures

In this section, we discuss the performance of the incremental clustering algorithms when different similarity measures—d_1, d_2, and d_∞—are used in the clustering functions. For this aim, we apply `DG-Clust` on different sizes of data sets: Bavaria postal 1, Bavaria postal 2, Iris plant, TSPLIB1060, Breast cancer Wisconsin, TSPLIB3038, D15112, Image segmentation, Page blocks, Pla85900, EEG eye state, and KEGG metabolic relation network. We use the cluster function values, the CPU time and Voronoi diagrams to compare results. The maximum number of clusters in extra small data sets is 10, in small size data sets 15 and it is 20 in all other data sets.

12.4.1 Optimal Values for Cluster Functions

Table 12.7 presents optimal values of the cluster function f_k obtained using similarity measures d_1, d_2, d_∞ and different number k of clusters. Note that these values are multiplied by m—the number of points in a data set—and also by numbers shown under names of data sets. The results show that in all cases, except

Iris plant data set, the values of the cluster function with d_∞ are the smallest among all three similarity measures.

12.4.2 Computational Time

The dependence of the CPU time used by `DG-Clust` for similarity measures d_1, d_2, and d_∞ are depicted in Fig. 12.21. The following conclusions can be made based on these results:

Table 12.7 Optimal values for cluster functions with different similarity measures

k	d_1	d_2	d_∞	d_1	d_2	d_∞	d_1	d_2	d_∞
	Bavaria postal 1			Bavaria postal 2			Iris plant		
	$\times 10^6$	$\times 10^{10}$	$\times 10^6$	$\times 10^6$	$\times 10^{10}$	$\times 10^6$	$\times 10^2$	$\times 10^2$	$\times 10^2$
2	4.0249	60.2547	3.9940	1.8600	5.2192	0.9456	2.1670	1.5235	0.9715
3	2.8284	29.4507	2.7892	1.2607	1.7399	0.6594	1.5920	0.7885	0.7420
5	1.7208	5.9762	1.6948	0.7872	0.5442	0.4221	1.2460	0.4645	0.5860
7	1.0704	2.1983	1.0368	0.5659	0.2215	0.2946	1.0620	0.3430	0.4915
10	0.6037	0.6447	0.5828	0.4340	0.1181	0.2173	0.9070	0.2583	0.4245
	TSPLIB1060			TSPLIB3038			Breast cancer		
	$\times 10^7$	$\times 10^9$	$\times 10^6$	$\times 10^6$	$\times 10^9$	$\times 10^6$	$\times 10^4$	$\times 10^4$	$\times 10^4$
2	0.3864	9.8319	2.6809	3.7308	3.1688	2.5651	0.6401	1.9323	0.1831
3	0.3139	6.7058	2.1508	3.0056	2.1763	2.1221	0.5702	1.6256	0.1607
5	0.2310	3.7915	1.6546	2.2551	1.1982	1.5576	0.5165	1.3707	0.1460
10	0.1563	1.7553	1.1048	1.5508	0.5634	1.0738	0.4270	1.0212	0.1278
15	0.1198	1.1219	0.8827	1.2295	0.3560	0.8592	0.3872	0.8711	0.1172
	D15112			Image segmentation			Page blocks		
	$\times 10^8$	$\times 10^{11}$	$\times 10^8$	$\times 10^6$	$\times 10^7$	$\times 10^5$	$\times 10^7$	$\times 10^{10}$	$\times 10^6$
2	0.8860	3.6840	0.6109	0.5192	3.5606	1.4929	0.8414	5.7937	4.1746
3	0.6908	2.5324	0.4896	0.4160	2.7416	1.3284	0.6747	3.3134	3.4309
5	0.4998	1.3271	0.3619	0.3400	1.7143	1.1081	0.4882	1.3218	2.4671
10	0.3618	0.6489	0.2524	0.2575	0.9967	0.8170	0.3152	0.4533	1.4446
15	0.2930	0.4324	0.2065	0.2188	0.6556	0.6966	0.2555	0.2495	1.1784
20	0.2501	0.3218	0.1768	0.1942	0.5137	0.6200	0.2200	0.1672	1.0160
	Pla85900			EEG eye state			KEGG metabolic		
	$\times 10^{10}$	$\times 10^{15}$	$\times 10^{10}$	$\times 10^7$	$\times 10^8$	$\times 10^6$	$\times 10^7$	$\times 10^8$	$\times 10^6$
2	2.0656	3.7491	1.4533	0.5289	8178.1381	1.5433	0.3586	11.3853	1.9821
3	1.6262	2.2806	1.1434	0.4197	1833.8806	0.9049	0.2800	4.9006	1.5112
5	1.2587	1.3397	0.8712	0.2944	1.3386	0.5183	0.2095	1.8837	1.0549
10	0.8950	0.6829	0.6218	0.2191	0.4567	0.3947	0.1459	0.6352	0.6667
15	0.7335	0.4625	0.5082	0.1965	0.3500	0.3562	0.1231	0.3512	0.5114
20	0.6374	0.3517	0.4443	0.1827	0.2899	0.3292	0.1108	0.2654	0.4440

- `DG-Clust` requires the largest CPU time with d_∞ in all data sets except Bavaria postal 1 and TSPLIB1060, and the least CPU time with d_2 in all data sets. The clustering problem with d_∞ is the most complex one and `DG-Clust` requires a large number of approximate subgradient evaluations to find search directions in this problem. On the other hand, the clustering problem with d_2 is the easiest

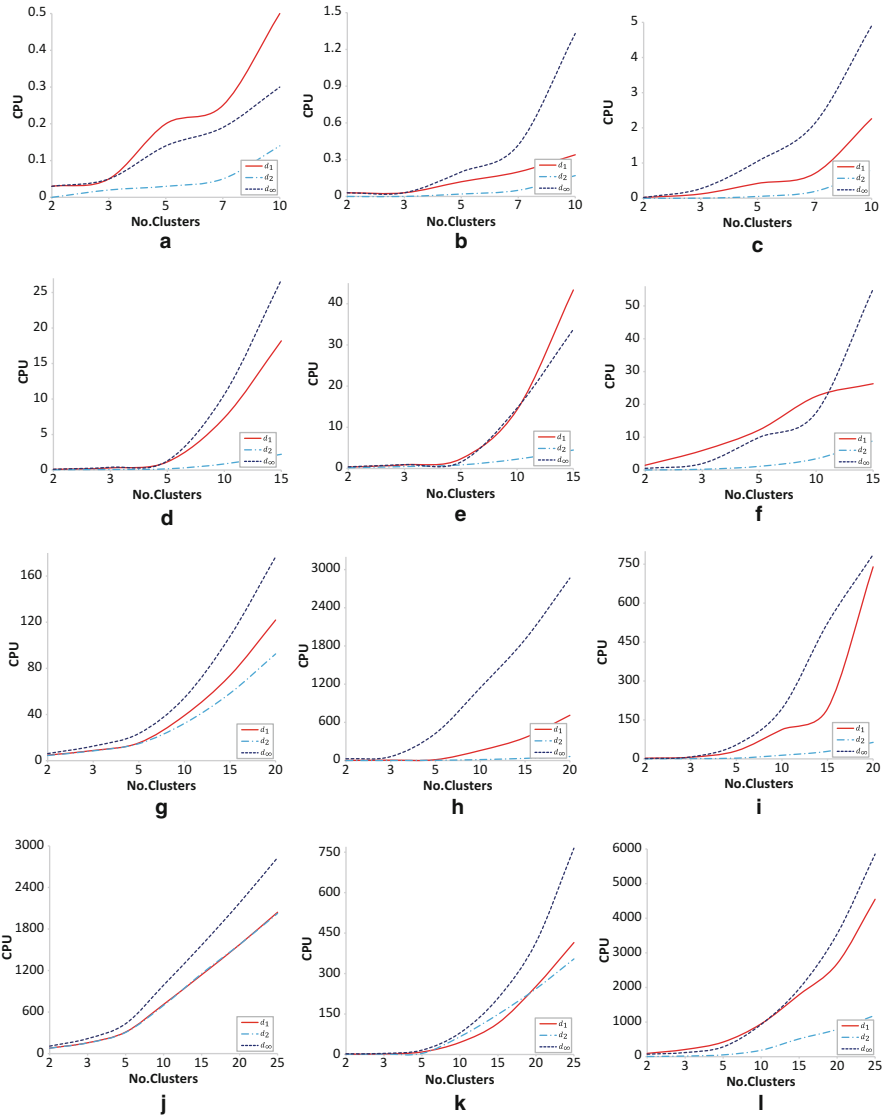

Fig. 12.21 CPU time with different similarity measures. (**a**) Bavaria postal 1. (**b**) Bavaria postal 2. (**c**) Iris plant. (**d**) TSPLIB1060. (**e**) TSPLIB3038. (**f**) Breast cancer. (**g**) D15112. (**h**) Image segmentation. (**i**) Page blocks. (**j**) Pla85900. (**k**) EEG eye state. (**l**) KEGG metabolic

as d_2 is smooth. In this case, the optimization method does not require a large number of approximate subgradient evaluations to find search directions;

- the CPU time required by `DG-Clust` depends more strongly on the number of attributes than on the number of data points. This claim is confirmed by comparing results for data sets with the similar number of data points and significantly different number of attributes: Image segmentation, TSPLIB3038, D15112, EGE eye state, Pla85900, and KEGG metabolic relation network. The comparison shows that `DG-Clust` becomes time-consuming in large data sets with the large number of attributes. In such data sets the size of the optimization problem increases rapidly as the number of clusters increase; and
- for all similarity measures the CPU time required at each iteration of the incremental algorithm, in general, is more than that of required at the previous iterations. This is due to the fact that the size of the optimization problem for finding all cluster centers increases at each iteration of the incremental algorithm.

12.4.3 Visualization of Results

Voronoi diagrams are used to visualize results obtained by `DG-Clust` in three data sets: German towns, TSPLIB1060 and TSPLIB3038. We utilize the software from [259] for this purpose. Figures 12.22, 12.23 and 12.24 present Voronoi diagrams for these data sets with five clusters. We can see that cluster structures for similarity measures d_1, d_2, and d_∞ are different in all data sets, although the distributions of cluster centers for d_1 and d_2 functions in TSPLIB1060 data set are similar.

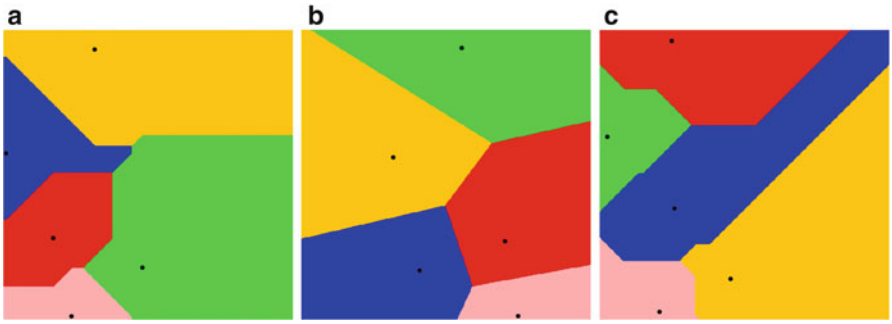

Fig. 12.22 Visualization of clusters in German towns data set. (**a**) L_1-norm. (**b**) L_2-norm. (**c**) L_∞-norm

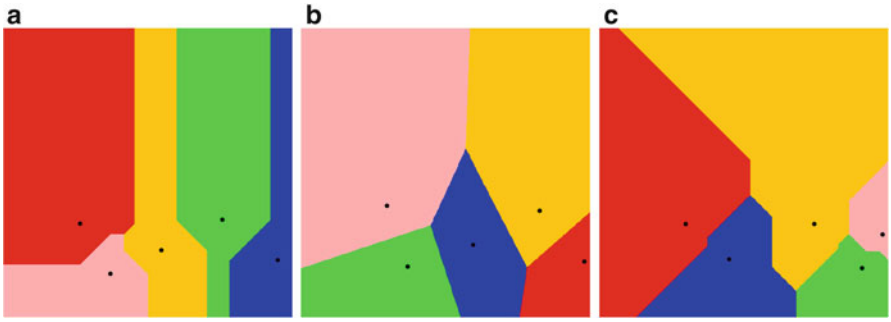

Fig. 12.23 Visualization of clusters in TSPLIB1060 data set. (**a**) L_1-norm. (**b**) L_2-norm. (**c**) L_∞-norm

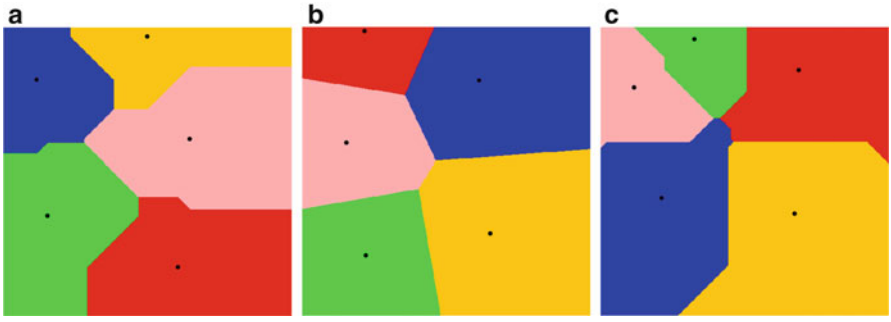

Fig. 12.24 Visualization of clusters in TSPLIB3038 data set. (**a**) L_1-norm. (**b**) L_2-norm. (**c**) L_∞-norm