STUDIA INFORMATICA 2004
Volume 25 Number 1 (57)

Marcin GORAWSKI, Ewa PŁUCIENNIK
Politechnika Śląska, Instytut Informatyki

COMPARATIVE ANALYSIS OF CLUSTERING ALGORITHMS IMPLEMENTED IN IBM INTELLIGENT MINER, ORACLE9I DATA MINING AND MICROSOFT ANALYSIS SERVICES

Summary. Current database systems development forces applying new information processing technologies. One of them is data mining, composed, among the others, of classification, clustering and association rules. So far, this technology has no general valid standards. Clustering is a special case of data mining – often it is an initial stage for applying other information analyses. In this article we present short comparison of clustering algorithms implemented in commercial tools produced by companies, which are very active in discovering new methods in the area of information processing.

Keywords: data mining, databases, clustering

ANALIZA PORÓWNAWCZA ALGORYTMÓW KLASTERYZACJI ZAIMPLEMENTOWANYCH W IBM INTELLIGENT MINER, ORACLE9I DATA MINING I MICROSOFT ANALYSIS SERVICES

Streszczenie. Obecny rozwój systemów baz danych wymusza stosowanie nowych technik przetwarzania informacji. Jedną z nich jest eksploracja danych, na którą składają się m.in. klasyfikacja, klasteryzacja czy reguły asocjacji. Technika ta jak do tej pory nie doczekała się ogólnie obowiązujących standardów. Klasteryzacja jest szczególnym przypadkiem eksploracji danych – często stanowi ona etap wyjściowy do stosowania pozostałych technik analizy informacji. Niniejszy artykuł przedstawia krótkie porównanie algorytmów klastrujących, zaimplementowanych w komercyjnych narzędziach firm, które bardzo aktywnie działają w obszarze nowych sposobów przetwarzania danych.

Słowa kluczowe: eksploracja danych, bazy danych, klasteryzacja

1. Introduction

Rapidly growing amount of data processed by an institution related to the world of business, industry and science, as well as the increasing importance of those data caused that the data processing has changed. Database systems are becoming more effective. New, intelligent information processing technologies have emerged.

To these technologies we can rate data mining identified with knowledge discovery systems. Data mining is defined as a part of a process of discovering implicit, previously unknown and potentially useful information, such as rules, relationships and regularities in databases.

Data mining elements are i.a. classification, clustering (unsupervised classification) and discovering association rules [2].

Although data mining has become inseparable element of data processing area, so far it has no general, valid standards [12].

Corporations like IBM, Oracle or Microsoft decide about data mining usefulness in applied information technology area.

IBM Intelligent Miner 6.1 has wide functionality range. It includes i.a. clustering, classification, association rules and prediction. Oracle9i has built in data mining mechanisms. Those mechanisms encompass clustering, classification and attribute importance model as well as the association rules. In Microsoft SQL Server 2000 data mining functionality appears in form of SQL Server Analysis Services integrated with OLAP services which is undoubtedly great merit. However Microsoft tool offers only two algorithms: clustering and decision trees. It has to be pointed out that all mentioned tools are adapted to exchange (export\import) some of the models in PMML (Predictive Model Markup Language) format [7, 12].

2. Clustering

Clustering is a very special data mining technique because it is often initial stage of data analysis. Clustering is a process of dividing data (abstract of physical objects) into groups with high intra-cluster similarity and low inter-cluster similarity. There are two approaches to this process: top-down regarding clustering as the segmentation of heterogeneous set into homogeneous groups and a bottom-up, which describes clustering as finding groups according to some natural criteria. Method of clustering strongly depends on the data characteristic. There is no universal cluster definition [4]. Because of above-mentioned

problems there is no universal clustering algorithm nor common definition of object similarity or cluster representation which can be applied on data irrespective of their domain, dimensionality or size.

Clustering lets user divide huge data set into smaller pieces, which then can be processed with another techniques as autonomous units. Clustering can be an initial stage for classification – we can assign adequate labels to obtained groups (for example in form of rules defining cluster membership). Clustering is a part of the unsupervised learning area – it does not require initial definition of group labels, predicates or target attributes.

3. Comparison of clustering algorithms implementations

Our goal was to compare clustering algorithms implemented in data mining commercial tools: IBM Intelligent Miner 6.1, Oracle9i Data Mining and Microsoft Analysis Services according to their results, not taking into account their scalability, performance or complexity.

We prepared three representative sets of 199 data points with coordinates range from –100 to 100 in 2D Euclidean space (for the purposes of easy visualization without the necessity to use specialized tools).

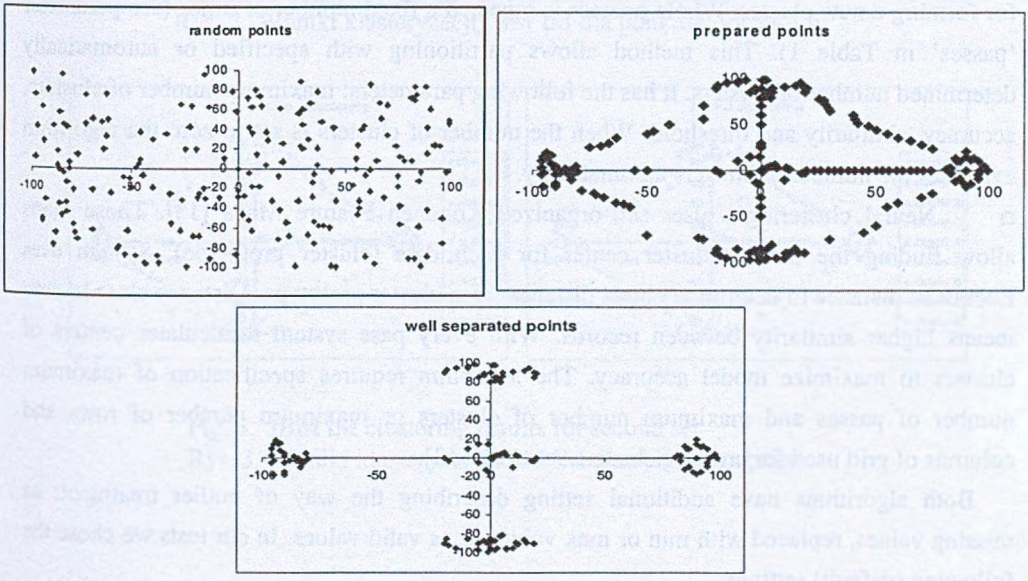

Fig. 1. Test data sets
Rys. 1. Testowe zbiory danych

First set consisted of randomly distributed points, second was built in order to emulate dense areas, outliers and noise chains in 2D space. Third set was created by eliminating noise from the second set in order to obtain well separated points areas (Figure 1). All points in the test sets had two numerical attributes – coordinates X and Y (K-means algorithm implemented in Oracle9i Data Mining does not allow categorical attributes).

4. IBM Intelligent Miner

IBM Intelligent Miner offers two clustering methods:

- Demographic clustering – similarities among records are determined by record fields' values comparison, clusters are created in a way that Condorcet criterion (the sum of all similarities in the pairs of records in the cluster minus the sum of all similarities in the pairs of records from various clusters) is maximized [13]. Algorithm builds clusters comparing record with all previously created clusters and assigning record to the cluster that maximizes a similarity score. Similarity between two objects is computed as a sum of votes for each pair of attributes values. Vote can range from –1 (different values) to 1 (identical values). Comparing a record with a cluster is accomplished by using value distribution of the cluster. In case the similarity score is negative for all existing clusters the record becomes a candidate for forming a new cluster. Whole process is repeated a fixed number of times (see parameter 'passes' in Table 1). This method allows partitioning with specified or automatically determined number of clusters. It has the following parameters: maximum number of clusters, accuracy, similarity and threshold. When the number of clusters is set to zero the algorithm evaluates the number of clusters automatically.
- Neural clustering – uses self-organized Kohonen Feature Maps [13]. These maps allow finding the closest cluster center for each data (cluster prototype). System uses Euclidean distance to determine object distance from cluster prototype. Distance close to zero means higher similarity between records. With every pass system recalculates centers of clusters to maximize model accuracy. The algorithm requires specification of maximum number of passes and maximum number of clusters or maximum number of rows and columns of grid used for creating clusters structure [13].

Both algorithms have additional setting describing the way of outlier treatment: as missing values, replaced with min or max values or as valid values. In our tests we chose the following (default) settings:

Table 1

IBM IM clustering algorithms parameters

Demographic clustering	Neural clustering
Maximum passes: 5 Maximum cluster: 9 and 0 Accuracy: 1.0 Similarity threshold: 0.5	Maximum passes: 5 Maximum rows: 3 Maximum columns: 3
Outlier treatment: treat outliers as missing values	

The following three figures show results of applying above algorithms to our test data sets.

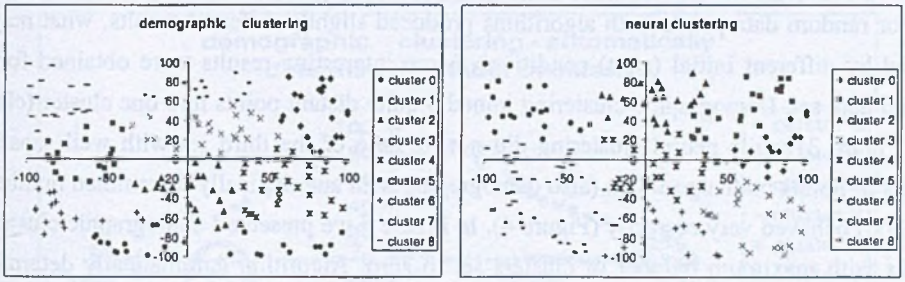

Fig. 2. IBM IM clustering results for random points
Rys. 2. Wyniki klasteryzacji IBM IM dla punktów losowych

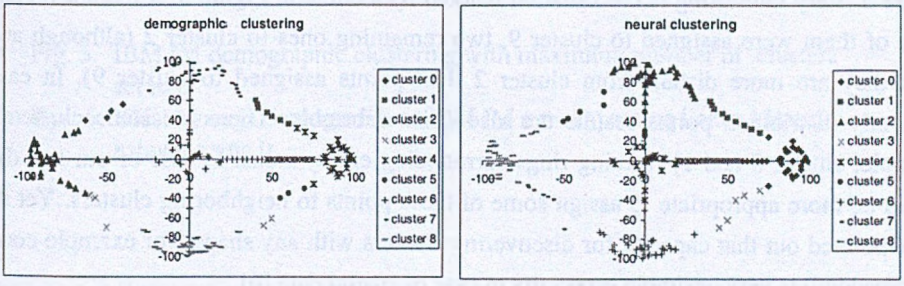

Fig. 3. IBM IM clustering results for second set
Rys. 3. Wyniki klasteryzacji IBM IM dla zbioru drugiego

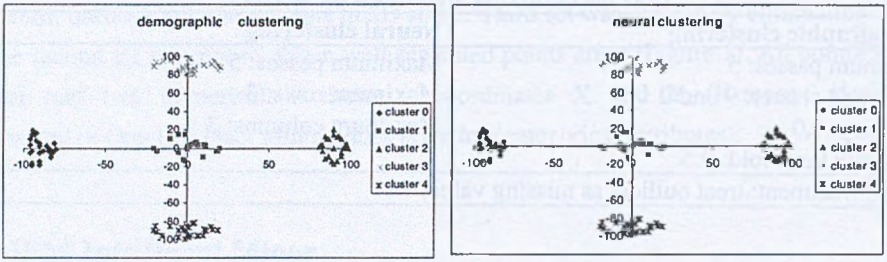

Fig. 4. IBM IM clustering results for third set
Rys. 4. Wyniki klasteryzacji IBM IM dla zbioru trzeciego

For random data points both algorithms produced slightly different results, what may be caused by different initial (start) conditions. More interesting results were obtained for the second data set. Demographic clustering joined 5 quite distant points into one cluster (cluster 3 in Figure 3) while neural clustering did not. In case of the third set with well separated groups of points both algorithms (also demographic with automatically determined number of clusters) behaved very correctly (Figure 4). In Figure 5 we presented demographic clustering results with maximum number of clusters set to zero. Algorithm automatically determined number of clusters to eleven for random points set and ten for prepared points set. Unfortunately, some of the clusters created for the prepared points were results of noise and outliers, i.e. cluster 3 (3 distant points), cluster 9 (4 distant points) or cluster 5 (3 points) in Figure 5. Very interesting is the situation marked by a circle in Figure 5. It includes 5 points, three of them were assigned to cluster 9, two remaining ones to cluster 2 (although at first sight they are more distant from cluster 2 then points assigned to cluster 9). In case of randomly distributed points results are also quite debatable. There are some clusters (for example, cluster 0 and 1) forming rings surrounding empty areas. In case of our test data it would be more appropriate to assign some of these points to neighboring clusters. Yet it has to be pointed out that capacity for discovering clusters with any shape (for example concave or embedded) is very desirable especially in case of spatial data [6].

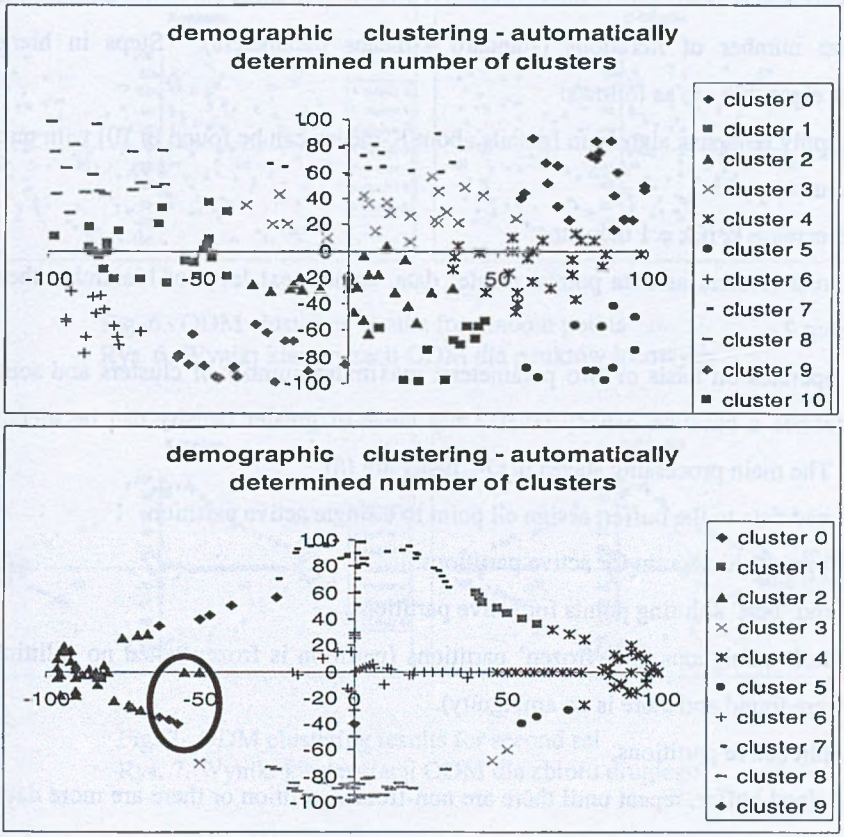

Fig. 5. IBM IM demographic clustering with maximum number of clusters set to 0

Rys. 5. Klasteryzacja demograficzna IBM IM z maksymalną liczbą klastrów ustawioną na 0

5. Oracle Data Mining

In Oracle Data Mining cluster is described by a centroid, attributes histogram and position in hierarchical model tree. Centroid is a vector that includes:

1. Mean value (for numerical attributes),
2. Mode value (for categorical attributes).

Oracle Data Mining implements two clustering algorithms: hierarchical version of the K-means algorithm and Oracle proprietary algorithm O-Cluster (hierarchical, grid based) [8].

The first one has three parameters: number of clusters, minimum error tolerance and the maximum number of iterations (standard k-means parameters). Steps in hierarchical K-means algorithm are as follows:

1. Apply K-means algorithm [details about K-means can be found in 10] with number of clusters = k.
2. Decrease k, if k = 1 then stop.
3. Treat clusters as data points, cluster data, create next level of hierarchy, then go to step 2.

Second operates on basis of two parameters: maximum number of clusters and sensitivity, which defines a baseline density level. Only areas of greater density can be identified as clusters. The main processing stages in O-Cluster are [8]:

1. Load data to the buffer, assign all point to a single active partition.
2. Compute histogram for active partitions.
3. Find 'best' splitting points for active partitions.
4. Mark ambiguous and 'frozen' partitions (partition is frozen when no splitting point were found and there is no ambiguity).
5. Split active partitions.
6. Reload buffer, repeat until there are non-frozen partition or there are more data points to process.

Cluster are used for generating Bayesian probability model during assigning new points to the cluster. In our tests we chose the following (default except for sensitivity) settings:

Table 2

Clustering algorithms parameters – ODM

K-means	**O-Cluster**
Number of clusters: 7 Minimum error tolerance: 0.005 Maximum iteration: 10	Maximum number of clusters: 7 Sensitivity: 0, 0.5 and 1

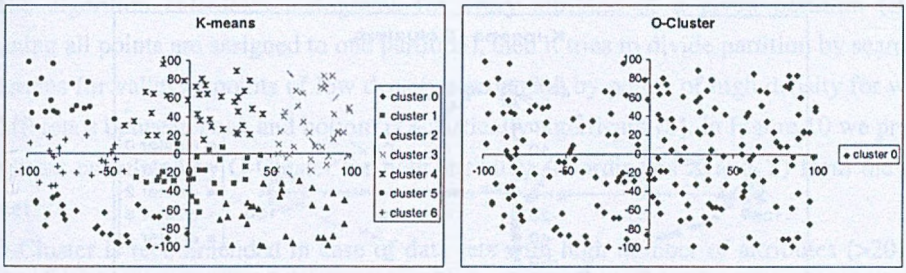

Fig. 6. ODM clustering results for random points
Rys. 6. Wyniki klasteryzacji ODM dla punktów losowych

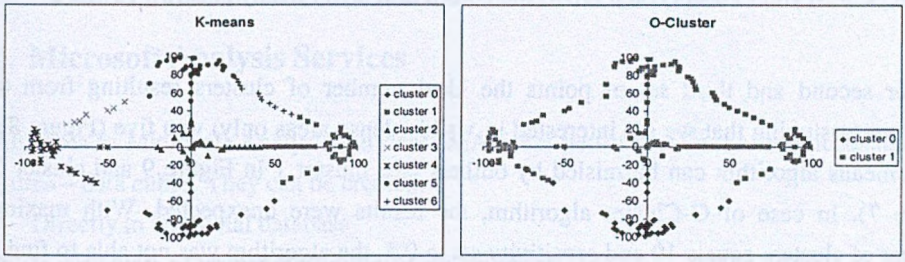

Fig. 7. ODM clustering results for second set
Rys. 7. Wyniki klasteryzacji ODM dla zbioru drugiego

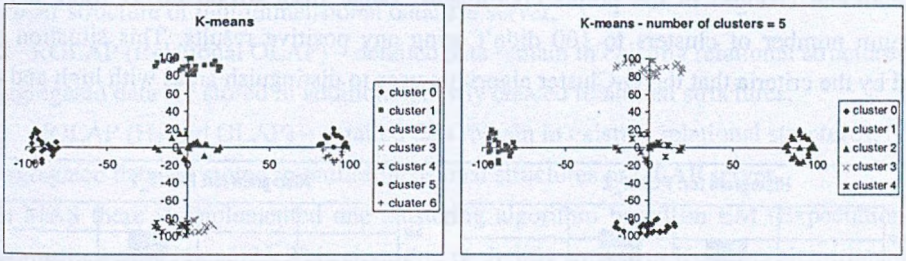

Fig. 8. ODM clustering results for third set
Rys. 8. Wyniki klasteryzacji ODM dla zbioru trzeciego

Results of applying K-means algorithm on the test data were quite expectable. For random points and number of clusters equal 4, the algorithm divided the points according to the coordinate system axes. For 7 clusters, points were assigned to groups containing from 19 up to 37 points.

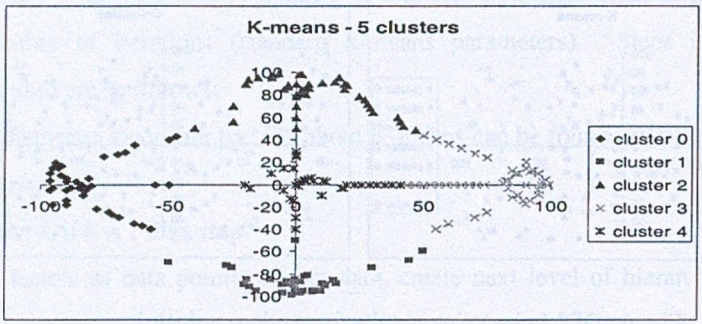

Fig. 9. ODM clustering results for second set and number of clusters set to 5
Rys. 9. Wyniki klasteryzacji ODM dla zbioru drugiego i liczby klastrów równą 5

For second and third set of points the ideal number of clusters resulting from their structure (assuming that we are interested in visibly dense areas only) was five (Figure 8 and 9). K-means algorithm can be misled by outliers (see cluster 1 in Figure 9 and cluster 0 in Figure 7). In case of O-Cluster algorithm, the results were unexpected. With maximum number of clusters equals 10 and sensitivity set to 0.5, the algorithm was not able to find any clusters in all data sets (more precisely – the algorithm found one cluster including all points). After setting sensitivity to the highest possible value (1) the situation slightly changed , but only for the second set (see Figure 7).

Additional changes in the points coordinates' range from –1000 up to 1000 and the maximum number of clusters to 100 didn't bring any positive results. This situation was caused by the criteria that the O-Cluster algorithm uses to distinguish areas with high and low density.

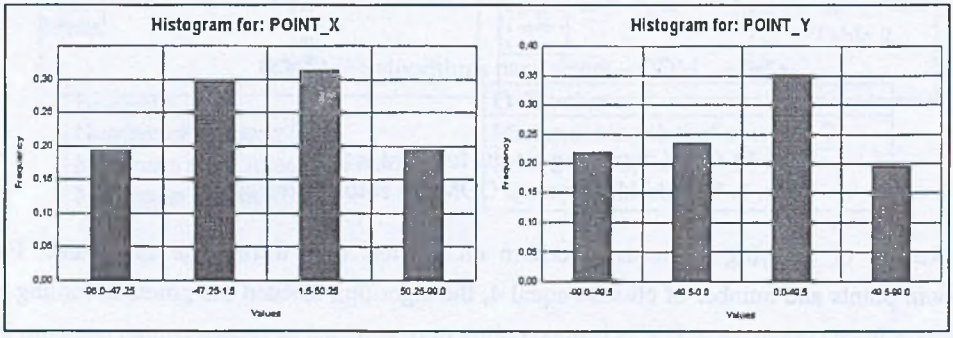

Fig. 10. Attribute histograms for the third set (O-Cluster) - ODM
Rys. 10. Histogramy atrybutów dla trzeciego zbioru danych (O-Cluster) - ODM

The algorithm calculates histograms for every attribute of a given partition (at the beginning all points are assigned to one partition), then it tries to divide partition by searching histograms for valleys - points of low density surrounded by points of high density for which the difference between peak and bottom is statistically significant [8]. In Figure 10 we present histograms calculated by O-Cluster for both attributes (coordinates X and Y) from the third data set.

O-Cluster is recommended in case of data sets with high number of attributes (>20) and records (>1000) [9]. Results of applying the algorithm for the two dimensional data set consisting of one hundred clusters with number of points ranging from 0 to 2000 are presented in [8].

6. Microsoft Analysis Services

Analyses in Microsoft Analysis Services (MAS) are performed on the multidimensional structures – data cubes. They can be created:

- Directly in relational database
- In structures of proper multidimensional database

Then the source data can be processed in OLAP (On-Line Analytical Processing) architecture of type [5]:

- MOLAP (Multidimensional OLAP) – detailed and aggregated data are stored in proper structure of multidimensional database server,
- ROLAP (Relational OLAP) – detailed data remain in existing relational structures, aggregated data are stored in additional, newly created relational structures,
- HOLAP (Hybrid OLAP) – detailed data remain in existing relational structures, aggregated data are stored in multidimensional structures of OLAP server.

In MAS there is implemented one clustering algorithm based on EM (Expectation and Maximization) algorithm [11]. This algorithm is slightly similar to K-means algorithm, it has one parameter (number of clusters), its characteristic feature is a 'soft' assigning points to clusters [1]. The algorithm iterates between two steps [11]:

1. Expectation – cluster membership calculations.
2. Maximization – subsequent estimation of the model parameters using cluster membership.

EM algorithm iteratively refines an initial cluster model to better fit the data until local optima is reached [1].

Results of applying the algorithm were the most difficult (among tested solution) in interpretation the cause being the assignment of same points to different clusters (Figure 11, 12 and 13).

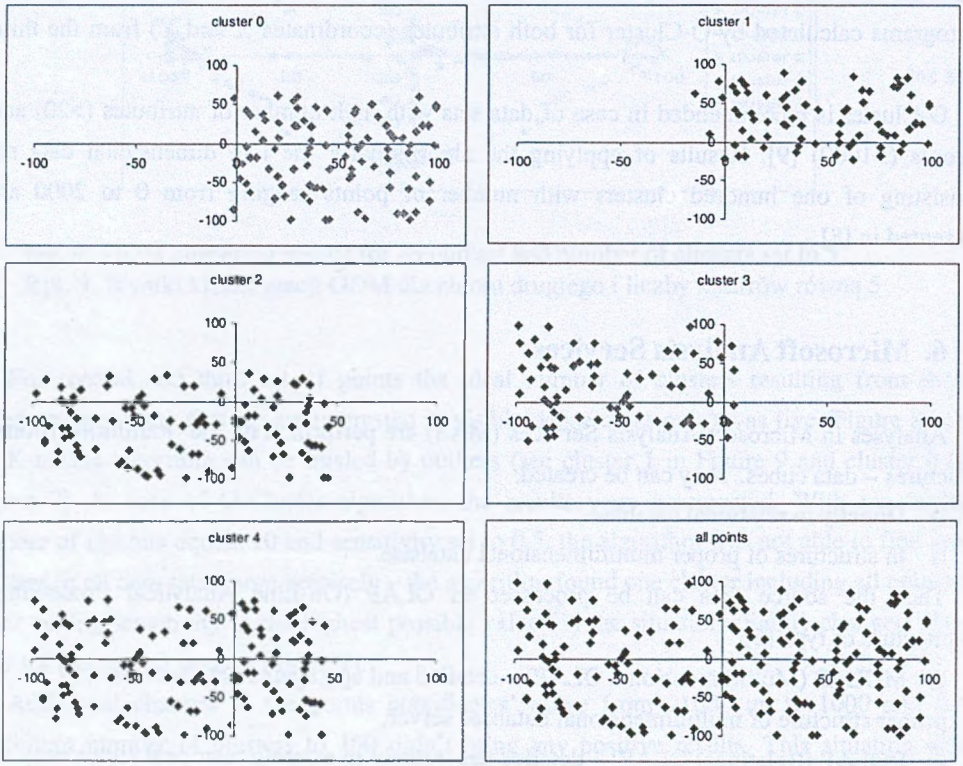

Fig. 11. MAS clustering results for random points
Rys. 11. Wyniki klasteryzacji MAS dla punktów losowych

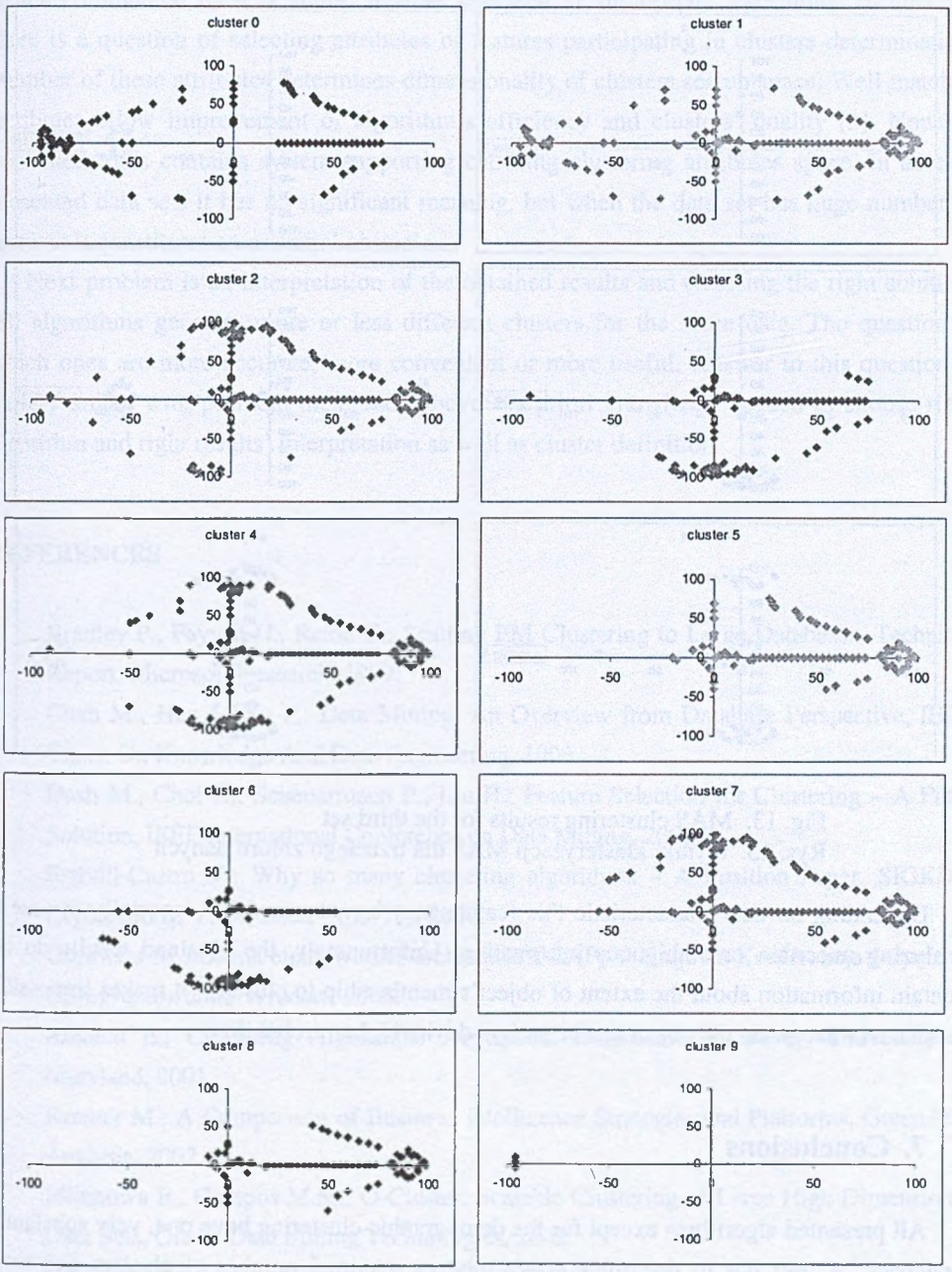

Fig. 12. MAS clustering results for the second set
Rys. 12. Wyniki klasteryzacji MAS dla drugiego zbioru danych

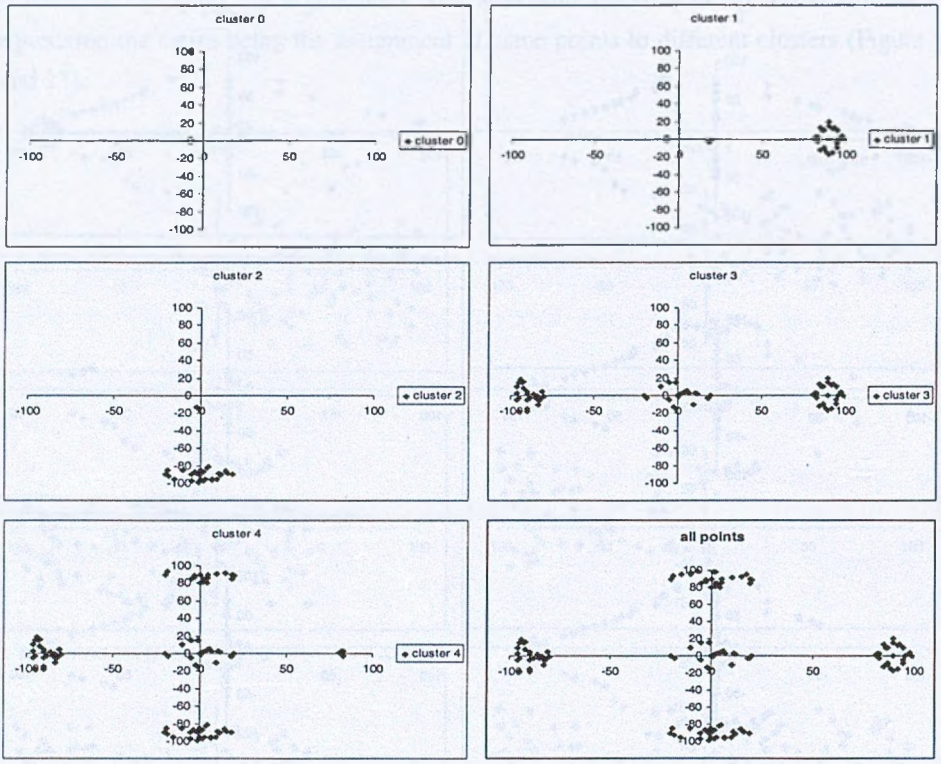

Fig. 13. MAS clustering results for the third set
Rys. 13. Wyniki klasteryzacji MAS dla trzeciego zbioru danych

Depending on data characteristic this feature may be very useful – especially in case of analyzing uncertain or ambiguous information. Unfortunately, the obtained results do not contain information about the extent of object's membership to cluster – it makes impossible to state to what group the object can be assigned with higher probability.

7. Conclusions

All presented algorithms except for the demographic clustering have one, very substantial drawback. A user has to determine either total or maximal number of clusters and some additional parameters. It means he or she should know, to some extent, the data characteristic. This fact stands in opposition to the idea of clustering, which is an unsupervised learning technique designated to analyze unknown data. There are algorithms supporting determination of optimal number of clusters (for example [10]), but applying them

in the commercial tools is slight. Besides selection of appropriate algorithms' parameters, there is a question of selecting attributes or features participating in clusters determination. Number of these attributes determines dimensionality of clusters search space. Well matched attributes allow improvement of algorithm's efficiency and clusters' quality [3]. None of presented tools contains system supporting choosing clustering attributes space. In case of presented data sets it has no significant meaning, but when the data set has huge number of features it constitutes an essential obstacle.

Next problem is an interpretation of the obtained results and choosing the right solution. All algorithms generate more or less different clusters for the same data. The question is which ones are more accurate, more convenient or more useful. Answer to this question is closely united with problem mentioned above – a priori knowledge needed to choose right algorithm and right results' interpretation as well as cluster definition.

REFERENCES

1. Bradley P., Fayyad U., Reina C.: Scalling EM Clustering to Large Databases, Technical Report, Microsoft Research, 1999.
2. Chen M., Han J., Yu P.: Data Mining: An Overview from Database Perspective, IEEE Trans. On Knowledge And Data Engineering, 1996.
3. Dash M., Choi K., Scheuermann P., Liu H.: Feature Selection for Clustering – A Filter Solution, IEEE International Conference on Data Mining, 2002.
4. Estivill-Castro V.: Why so many clustering algorithms – A Position Paper, SIGKDD Explorations, Vol.4, Issue 1 (65-75), 2002.
5. Gorawski M.: Ocena efektywności architektur OLAP, V Krajowa Konferencja *Inżynieria Oprogramowania,* Wrocław 2003.
6. Kolatch E.: Clustering Algorithms for Spatial Databases: A Survey, University of Maryland, 2001.
7. Kramer M.: A Comparison of Business Intelligence Strategies and Platforms, Green Hill Analysis, 2002.
8. Milenowa B., Campos M.M.: O-Cluster: Scalable Clustering of Large High Dimensional Data Sets, Oracle Data Mining Technologies, 2002.
9. Oracle 9i Data Mining Concepts, Release 2 (9.2), 2002.
10. Ray S., Turi R.H.: Determination of Number of Clusters in K-Means Clustering and Application in Colour Image Segmentation, Monash University, 1999.
11. Soni S., Tang Z.,Yang J.: Performance Study of Microsoft Data Mining Algorithms, Microsoft Corporation, 2001.

12. Świerzowicz J.: Impact of data mining standarization on information technology development, Studia Informatica 2003, Vol.24, Nr 2A (53).
13. Using the Intelligent Miner for Data, Version 8 Release 1, IBM, 2002.

Recenzent: Dr inż. Janusz Świerzowicz

Wpłynęło do Redakcji 1 październik 2003 r.

Omówienie

Obecny rozwój systemów baz danych wymusza stosowanie nowych technik przetwarzania informacji. Jedną z nich jest eksploracja danych, na którą składają się m.in. klasyfikacja, klasteryzacja czy reguły asocjacji. Technika ta jak do tej pory nie doczekała się ogólnie obowiązujących standardów. Klasteryzacja jest szczególnym przypadkiem eksploracji danych – często stanowi ona etap wyjściowy do stosowania pozostałych technik analizy informacji.

Celem niniejszej pracy było porównanie algorytmów klasteryzacji zaimplementowanych w komercyjnych narzędziach IBM w Intelligent Miner 6.1, Oracle Data Mining oraz SQL Server Analysis Services wg kryterium oceny wyników. Wykorzystano w tym celu trzy zbiory danych: losowo wybranych punktów; punktów emulujących występowanie gęstych obszarów danych, elementów zakłóconych oraz łańcuchów zakłóceń; dobrze separowalnych skupisk punktów (rys. 1).

IBM Intelligent Miner udostępnia dwie metody klasteryzacji: klasteryzację demograficzną oraz neuronową. W naszych testach posłużyliśmy się domyślnymi ustawieniami (tabela 1). Rysunki 2, 3, 4 pokazują wyniki zastosowania powyższych algorytmów na naszych danych testowych. Rysunek 5 pokazuje wyniki klasteryzacji demograficznej z maksymalną liczbą klastrów ustawioną na zero.

W Oracle Data Mining zaimplementowano dwa algorytmy klasteryzacji: hierarchiczną wersję algorytmu K-średnich oraz algorytm własny Oracle'a O-Cluster (hierarchiczny, oparty o siatkę). Parametry powyższych algorytmów i ich wartości wykorzystane w testach zostały pokazane w tabeli 2. Uzyskane rezultaty zaprezentowana na rys 6, 7, 8 oraz 9.

W Microsoft Analysis Services zaimplementowano jeden algorytm klasteryzacji oparty na algorytmie EM (Expectation and Maximization). Wyniki jego działania były najtrudniejsze (pośród testowanych rozwiązań) do interpretacji ze względu na fakt przypisania tych samych punktów do różnych klastrów (rys. 11, 12 i 13).

Wszystkie zaprezentowane algorytmy z wyjątkiem klasteryzacji demograficznej posiadają jedną podstawową wadę - użytkownik musi określić całkowitą lub maksymalną liczbę klastrów oraz pewne dodatkowe parametry. Kolejnym problemem jest interpretacja otrzymanych wyników i wybór najlepszego rozwiązania, co bezpośrednio wiąże się z posiadaniem wstępnej wiedzy o dziedzinie danych potrzebnej do wybrania właściwego algorytmu oraz prawidłowej interpretacji uzyskanych rezultatów.

Adresy

Marcin GORAWSKI: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-101 Gliwice, Polska, M.Gorawski@zti.iinf.polsl.gliwice.pl .

Ewa PŁUCIENNIK: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16, 44-101 Gliwice, Polska, e.pluciennik@zti.iinf.polsl.gliwice.pl .