ACTA UNIVERSITATIS LODZIENSIS
FOLIA OECONOMICA 206, 2007

*Jerzy Korzeniewski**

COMPARATIVE ASSESSMENT OF SOME SELECTED METHODS OF DETERMINING THE NUMBER OF CLUSTERS IN A DATA SET

Abstract. This paper is an attempt to compare the performance of an algorithm for determining the number of clusters in a data set proposed by the author with other methods of determining the number of clusters. The idea of the new algorithm is based on the comparison of pseudo cumulative distribution functions of a certain random variable. For a fixed window size we draw K different points and for every point we find the corresponding limiting point in the mean shift procedure. Then we check if the distance (e.g. Euclidean) between every pair of the limiting points is greater than the window size. Analogously we determine the pseudo cumulative distribution functions for different numbers K of clusters. Out of all pseudo cumulative distribution functions we pick the proper one i.e. the "last one" (with respect to K) which has a horizontal phase. Other methods of determining the number of clusters in a data set are compared with the proposed algorithm in a number of examples of two dimensional data sets for different clustering methods (k-means clustering and minimum distance agglomeration).

Key words: cluster analysis, number of clusters, computer algorithm, mean shift method.

1. IDEA OF NEW ALGORITHM

The new algorithm is based on the sample mean shift method used to estimate the local maxima of the density function of a random vector. The idea of this method proposed by D. Comaniciu and P. Meer (1999) is as follows. Let $\{x_i\}_{i=1,\dots,n}$, be a set of n points from d-dimensional Euclidean space. The kernel estimator of multivariate density function with kernel $K(x)$ and window size h is given by the formula

$$\bar{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \qquad (1)$$

[*] Ph.D., Department of Statistical Methods, University of Łódź.

The optimal kernel in the sense of minimum square error is the Epanechnikov kernel given by the formula

$$K_E(x) = \begin{cases} 0.5c_d^{-1}(d+2)(1-x^{\mathrm{T}}x), & \text{if } x^{\mathrm{T}}x < 1 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where c_d is the volume of a unit sphere in d-dimensional Euclidean space. It is easy to find an estimator of the gradient of the density function

$$\nabla\bar{f}(x) = \frac{1}{nh^d}\sum_{i=1}^{n}\nabla K\left(\frac{x-x_i}{h}\right) \tag{3}$$

For the Epanechnikov kernel we will arrive at the formula

$$\nabla\bar{f}(x) = \frac{1}{nh^d c_d}\frac{d+2}{h^2}\sum_{x_i\in S_h(x)}[x-x_i] = \frac{n_x}{nh^d}\frac{d+2}{h^2}\left(1/n_x\sum_{x_i\in S_h(x)}[x-x_i]\right) \tag{4}$$

The quantity

$$M_h(x) = (1/n_x\sum_{x_i\in S_h(x)}[x-x_i]) = 1/n_x\sum_{x_i\in S_h(x)}x_i - x \tag{5}$$

is called the window/sample mean shift. The mean shift always moves the sample in the direction of the greatest increase in density, therefore, if we keep on moving the sample by the vector given by formula (5) we will get convergence towards the centre of the local density maximum (see: Comaniciu, Meer (1999)). By the limiting point of a given starting point we will understand the centre of the last window in the sequence of the mean shift procedures.

In connection with the algorithm proposed below it is important to remark that the window is shifted at every step of the procedure in the direction of the nearest local density maximum. The location of this maximum depends on the size h of the window. The smaller the value of h the more local is the character of the maximum, the greater the value of h the more global is the maximum. In particular, if the window size h is greater than the greatest distance between any two data points every data point will be shifted towards the same limiting point.

Formally, the algorithm can be described in the following steps.

Step 1. For $K = 2$ we draw dependently 2 data points and for each point we find the corresponding limiting point in the mean shift procedure for a fixed window size h.

Step 2. We check if among all pairs of limiting points (for $K = 2$ there is only one pair) there exists at least one pair of points with the distance smaller than h.

Step 3. We repeat step 1 and step 2 10000 times in order to find the probability of meeting the condition from step 2.

Step 4. We repeat steps 1, 2 and 3 for all window sizes h from interval (0, max. distance) with h increasing discreetly by small increments e.g. 1/1000 of the maximal distance. As a result we get a pseudo cumulative distribution function for $k = 2$.

Step 5. We repeat steps 1, 2, 3 and 4 for $K = 3, 4, 5, ...$, (e.g.) 20.

The proper number of clusters that is picked up with the help of the above presented algorithm is the one equal to the greatest K that corresponds to the curve possessing a "horizontal phase" significantly below than 1. Horizontal phase is defined in the following way: it is a part of the curve of the length of at least 1/20 of the median of all distances between pairs of points and each point of this part corresponds to a probability smaller or greater by no more than 0.01 than the probabilities for all other points from the part preceding the point. The numbers 1/20 of the median and 0.01 were found by the method of trial and error and obviously are not to be changed – are supposed to be working for an arbitrary data set. The horizontal phases are usually very evident and if the numbers 1/20 and 0.01 were slightly different it wouldn't change the algorithm's performance. The appearance of the median of all pairwise distances makes it necessary to estimate it. The following way of estimating it was adopted. If the data set has less than 200 elements we compute all pairwise distances and pick up the median. If the set is larger we draw without replacement 300 pairs of elements and take the median of the found 300 pairwise distances. The idea behind this algorithm is as follows.

Let us consider a two dimensional data set (see Fig. 1) consisting of three equally spaced identical unimodal clusters – each cluster centre e.g. 80 pixels away from each of the other clusters. Every drawn point will be shifted in the mean shift procedure to the very centre of its cluster because the cluster density increases with getting close to cluster's centre. Therefore, if we draw 2 points the probability of meeting the step 2 condition is equal to the probability of drawing 2 points from the same cluster and should stay constant no matter if the window size h is equal to 20, 30 or 70 pixels. If the window size exceeds 80 pixels the probability jumps to 1 on a short segment of the horizontal axis because all set points (including the 2 drawn) correspond to the same limiting point. Similar situation will take place in the case of drawing 3 points with the horizontal phase (the constant probability) being obviously higher. When we draw 4 points the probability of meeting step 2 condition has to be equal to 1 even for very small window sizes because some 2 points have to belong to the same cluster and therefore have the same limiting point. From the graph presenting the curves for the considered data set it is evident why we should pick the curve that is the last to possess the horizontal phase. The length of the horizontal phase is connected with the distance between the clusters' centres and the height on which the horizontal phase is placed is connected with the number of points in the cluster due to which the phase is created.

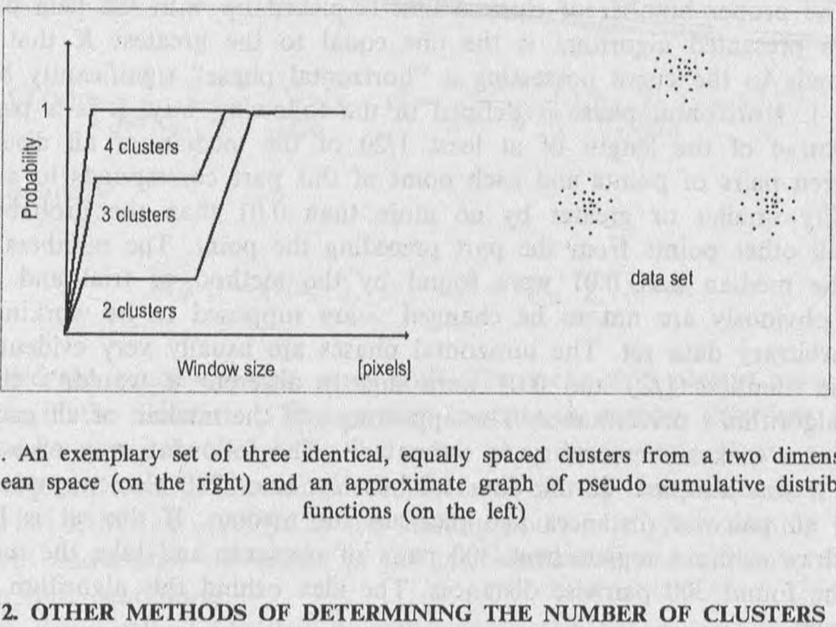


Fig. 1. An exemplary set of three identical, equally spaced clusters from a two dimensional Euclidean space (on the right) and an approximate graph of pseudo cumulative distribution functions (on the left)

2. OTHER METHODS OF DETERMINING THE NUMBER OF CLUSTERS

There is some difficulty in comparing the algorithm described in the previous section with other methods of determining the number of clusters in a data set because all methods which can be found in literature determine the optimal number of clusters for a given clustering method. We chose four methods whose performance is better than that of other methods (Sugar, James, 2003). In the following formulae K denotes the number of clusters which have to be constructed by some method, $B(K)$ and $W(K)$ denote, respectively, the between and within cluster variance. The first method is the Caliński–Harabasz index for which we should choose K that maximizes the value given by the formula

$$CH(K) = \frac{B(K)/(K-1)}{W(K)/(n-K)} \tag{6}$$

The second method is the Krzanowski–Lai index given by formula (7)

$$KL(K) = \left|\frac{DIFF(K)}{DIFF(K+1)}\right| \tag{7}$$

where

$$DIFF(K) = (K-1)^{2/d}W(K-1) - K^{2/d}W(K) \tag{8}$$

and again we should seek K that maximizes this index. The third method is the Hartigan index given by formula (9)

$$H(K) = (n-K-1)\left(\frac{W(K)}{W(K+1)} - 1\right) \tag{9}$$

in connection with which we should choose smallest K for which the index is smaller or equal to 10. The forth method is based the silhouette index which for the i-th element is given by the formula

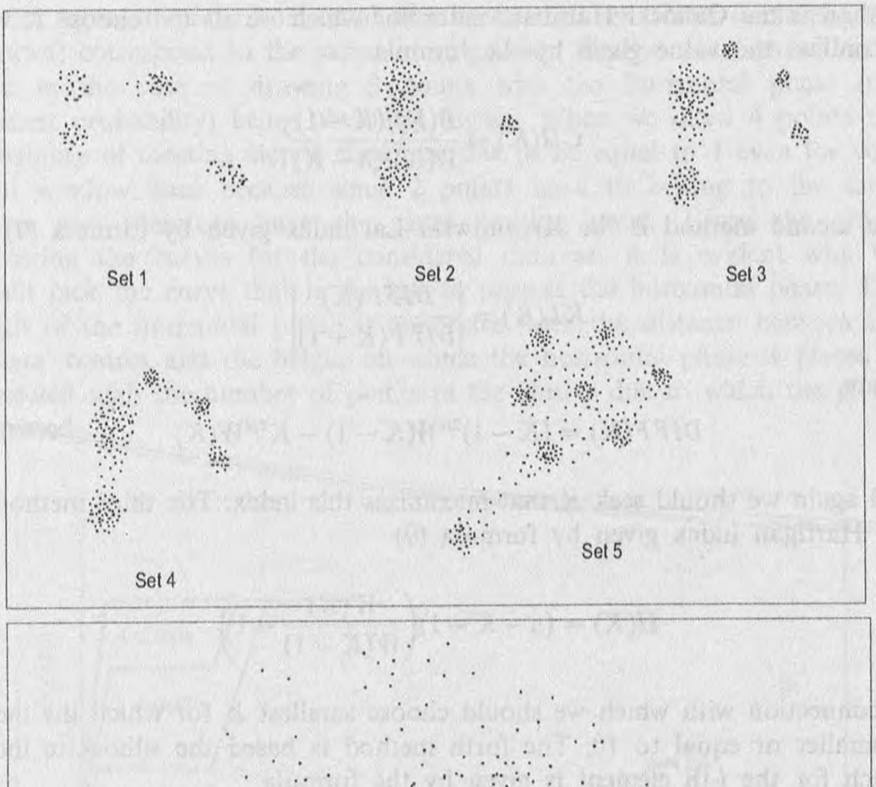
$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{10}$$

where $a(i)$ is the average distance between the i-th element and all other points in its cluster $b(i)$ is the average distance to points in the nearest cluster. We should choose K that maximizes the average value of $s(i)$.

All four methods will be checked for two quite different in nature clustering methods i.e. k-means clustering and nearest neighbour agglomeration algorithm.

3. PERFORMANCE ANALYSIS

We will try to compare how all five methods perform for six different two dimensional data sets. The sets were either created or chosen so as to represent well separated clusters, badly separated clusters, clusters with similar numbers of elements and clusters with different numbers of elements.



Set 6 data set, 300 points generated from three normal distributions

Fig. 2. Six investigated two dimensional sets of points

Source: sets 1–5 – own constructions, set 6 – Gordon 1999.

Table 1

Numbers of clusters as shown by the four compared indices for the six analysed data sets

Set	*k*-means clustering				Minimum distance agglomeration			
	Silhouette index	Caliński–Harabasz	Hartigan	Krzanowski–Lai	Silhouette index	Caliński–Harabasz	Hartigan	Krzanowski–Lai
1	2	7	⩾8	5	⩾7	5	5	3
2	3	5	⩾7	5	⩾7	3	3	3
3	5	8	⩾9	4	⩾8	5	5	5
4	4	7	⩾8	7	⩾8	5	3	5
5	10	10	⩾12	10	6	4	4	4
6	4	6	⩾7	2	3	5	2	2

Source: own calculations.

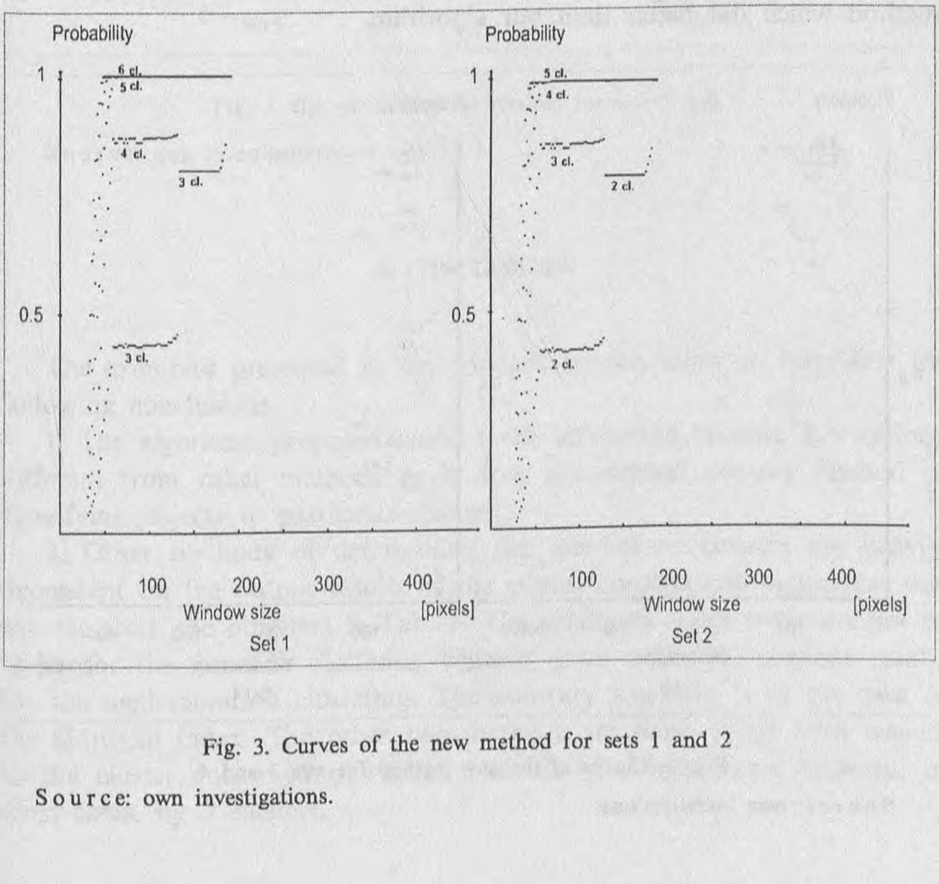


Fig. 3. Curves of the new method for sets 1 and 2

Source: own investigations.

For Set 1 which consists of quite evident 4 clusters the algorithm proposed showed unquestionable 4 clusters. However, for this seemingly easy to handle set all other methods give wrong indications for both clustering methods.

For Set 2 which consists of 3 (rather than 2) clusters the algorithm proposed showed the proper number, all other methods performing rather poorly.

For Set 3 which consists of 5 (rather than 4) clusters the algorithm proposed showed the proper number, all other methods performing so-so.

For Set 4 which consists of 5 (rather than 4) clusters, differing from the previous set only in indistinct borders between clusters, the algorithm proposed showed the proper number, all other methods performing badly or very badly.

For Set 5 which consists of 8 clusters the algorithm proposed showed 7 clusters all other methods performing very badly.

For Set 6 which is a very fuzzy set and which can be described as consisting of 3 or 4 clusters the algorithm proposed showed 4 or 5 clusters, all other methods performing very poorly apart from the silhouette index method which did better than our algorithm.

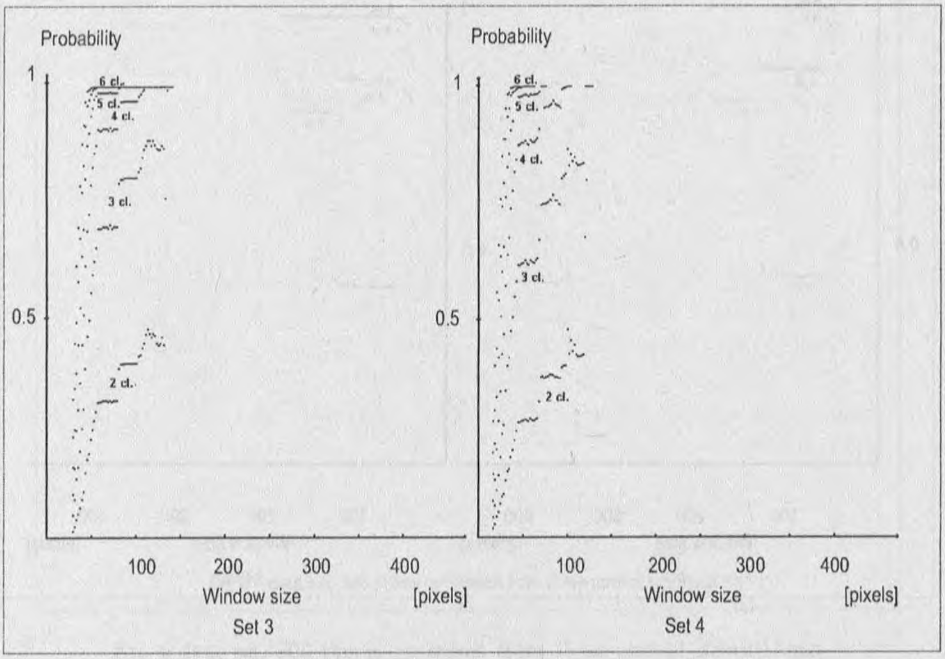


Fig. 4. Curves of the new method for sets 3 and 4

Source: own investigations.

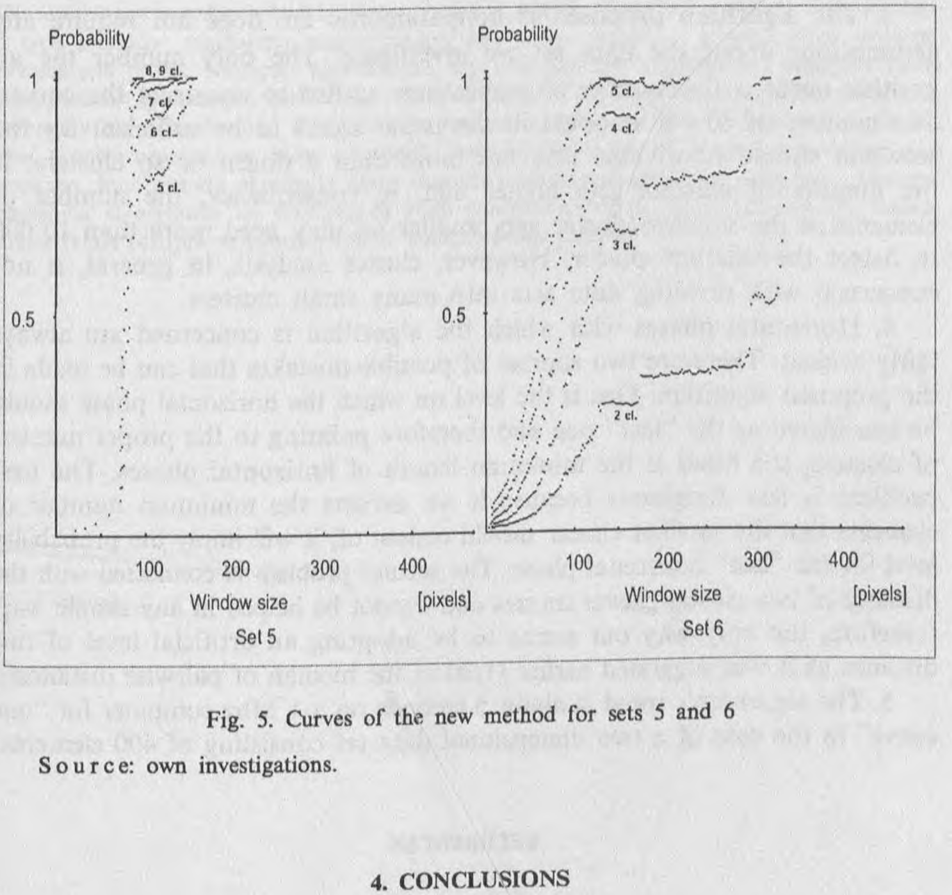


Fig. 5. Curves of the new method for sets 5 and 6

Source: own investigations.

4. CONCLUSIONS

The examples presented in the previous section allow to formulate the following conclusions.

1. The algorithm proposed seems to be interesting because it is entirely different from other methods as it does not depend on any method of classifying objects to particular clusters.

2. Other methods of determining the number of clusters are heavily dependent on the output results of the cluster construction method as one can see from the numbers in Tab. 1. The silhouette index performs not so badly for the *k*-means clustering while it gives entirely erroneous results for the agglomeration clustering. The contrary situation is in the case of the Hartigan index. The other two methods are more stable with respect to the cluster construction method but still give indications different, in most cases, by 2 clusters.

3. The algorithm proposed is nonparametric i.e. does not require any assumptions about the data set we investigate. The only number the algorithm needs is the number of replications needed to construct the curves. The number of 10 000 adopted in the paper seems to be sufficient for the sets that should be divided into not more than a dozen or so clusters. If the number of clusters gets higher and, in consequence, the number of elements in the smallest cluster gets smaller we may need more than 10 000 to detect the smallest cluster. However, cluster analysis, in general, is not concerned with dividing data sets into many small clusters.

4. Horizontal phases with which the algorithm is concerned are always fairly evident. There are two sources of possible mistakes that can be made in the proposed algorithm. One is the level on which the horizontal phase should be considered as the "last" one and therefore pointing to the proper number of clusters, the other is the minimum length of horizontal phases. The first problem is less dangerous because if we assume the minimum number of elements that the smallest cluster should consist of, it will imply the probability level for the "last" horizontal phase. The second problem is connected with the distance of two closest cluster centres and cannot be helped in any simple way, therefore, the only way out seems to be adopting an artificial level of this distance as it was suggested earlier (1/20 of the median of pairwise distances).

5. The algorithm's speed is about 5 seconds on a 1 Mhz computer for "one curve" in the case of a two dimensional data set consisting of 400 elements.

REFERENCES

Comaniciu D., Meer P. (1999), *Mean Shift Analysis and Applications*, IEEE Int. Conf. Computer Vision (ICCV'99), Kerkyra, Greece, 1197–1203.

Gordon A. D. (1999), *Classification*, Chapman & Hall, New York.

Sugar C. A., James G. M. (2003), *Finding the Number of Clusters in a Dataset: An Information – Theoretic Approach*, JASA, **98**, 750–763.

Jerzy Korzeniewski

OCENA PORÓWNAWCZA WYBRANYCH METOD WYZNACZAJĄCYCH ILOŚĆ SKUPIEŃ W ZBIORZE DANYCH

Artykuł niniejszy jest próbą oceny porównawczej algorytmu wyznaczającego ilość skupień w zbiorze danych, zaproponowanego przez autora, z innymi metodami wyznaczania ilości skupień. Algorytm autora oparty jest na porównaniu pseudodystrybuant pewnej zmiennej losowej dla różnych ilości skupień. Ta zmienna losowa jest zdefiniowana w następujący sposób.

Dla ustalonego rozmiaru okna losujemy ze zbioru danych *K* różnych punktów i dla każdego z tych punktów znajdujemy odpowiadający mu punkt graniczny w procedurze średniego przesunięcia próby. Następnie sprawdzamy, czy odległość (np. euklidesowa) pomiędzy każdą parą punktów granicznych jest większa od rozmiaru okna. Analogicznie wyznaczamy pseudodystrybuanty dla różnych ilości *K* skupień. Ze wszystkich dystrybuant za prawidłowo określającą ilość skupień uznajemy tę, która odpowiada ostatniej (względem *K*) krzywej, posiadającej fazę poziomą. Inne metody określania liczby skupień w zbiorze danych są porównane z zaproponowanym algorytmem na przykładach kilku dwuwymiarowych zbiorów danych dla dwóch, diametralnie różnych w naturze, metod konstruowania skupień.