A C T A   U N I V E R S I T A T I S   L O D Z I E N S I S
FOLIA OECONOMICA 3(302), 2014

Andrzej Dudek[*]

DYNAMIC CLASSIFICATION OF GEOGRAPHIC POINTS ON GOOGLE MAPS

Abstract. Classification of geographical points on Google maps is an interesting example of the use of cluster analysis algorithm in which the final number of clusters is obtained not only by presuppositions and the algorithm used, but also by the scale, on which map is actually displayed. The ultimate goal of classification is not only to obtain relatively homogeneous clusters, but also to prevent the phenomenon of "blurring" partitions on the map. In the paper there is proposed an algorithm that automatically creates a hierarchical structure of classes (which differs, however, from the structures obtained by the hierarchical agglomerative methods), in such way that the final classification takes into account the enlargement in which the map is displayed. The aim of article is illustrated with real examples on Google maps using JavaScript / JQuery.
Keywords: Cluster analysis, Google maps.

I. INTRODUCTION

Google maps technology and related such as *Openstreet maps* have been developed rapidly for several years, an increasing number of computer applications use maps for data presentation. In practice, in modern ERP systems (see for example [Dudek (ed.) 2011]), in addition to the traditional ways of listing business partners, it is possible to visualize them on maps and achieve additional operations such as route planning for sales representatives. This functionality is very useful feature of modern information systems, but it is fraught with a certain flaw that Google itself is called "too many clusters". It comes to a situation when the number of markers (corresponding to the addresses on a map described by geographical latitude and longitude) is too large, and presented the data is almost completely unreadable (see fig. 1).

The article proposes a modification of the classic algorithm of k-means, taking into account the distance between points that are described by the longitude and latitude, describes the relationship between scale in which the map is displayed and the structure of clusters and proposes algorithm creating different clusters structure for different scale map. The whole is completed with a short summary and open issues.

[*] Ph.D., Wrocław University of Economics, Department of Econometrics and Computer Science, andrzej.dudek@ue.wroc.pl.

Figure 1. „Bluring” of markers on map

Source: own development with usage of Google maps.

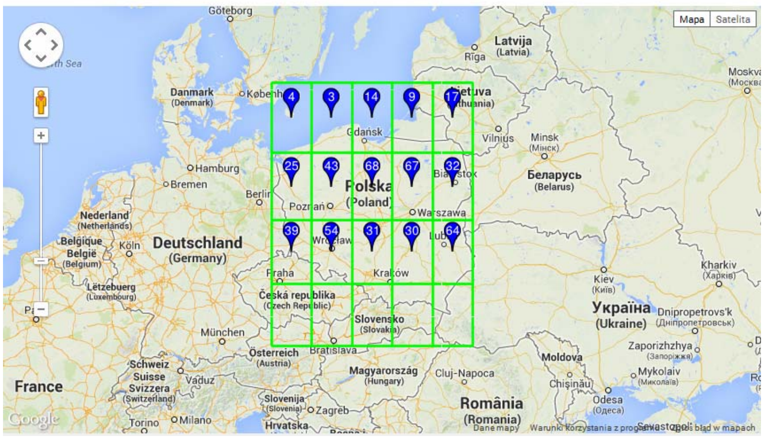
II NAIVE CLASSIFICATION METHODS FOR GOOGLE MAP MARKERS

The problem of unreadable markers on a map by too much density of markers was noticed by Google. In the document ["Too many markers"] it is proposed to apply in such a case, the three procedures called "classification procedures", though these are not classic cluster analysis algorithms and we can give them the name of "simple" or "naïve" methods. They are:

- Grid –based clustering
- Administrative units clustering
- Zip codes or telephone prefixes clustering

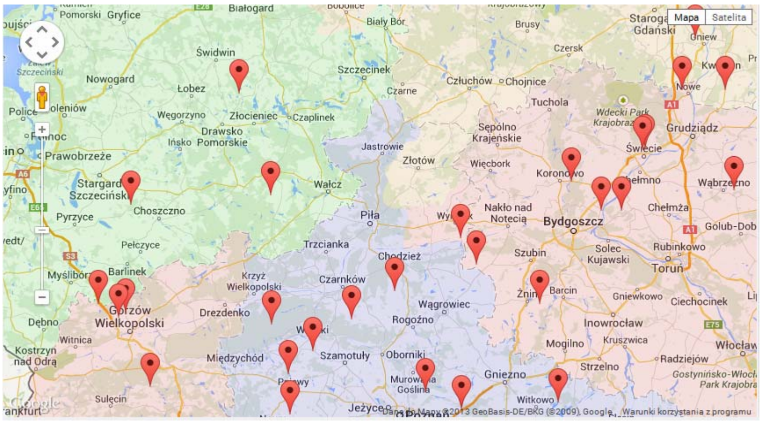
Fig. 2. presents examples of the use of these methods for markers of fig. 1. (a. and b.) and actual customers of real company (c. and d.)

a)



b)

c)



d)

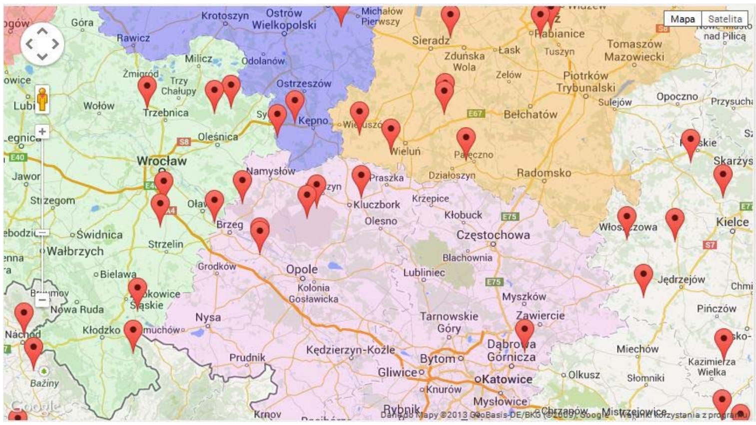


Figure 2. Naive classification methods for Google map markers

a), b) grid-based clustering; c) administrative units clustering – the various grey tones correspond to voivodships; d) telephone prefixes clustering - various shades of grey correspond to enterprises within the same group of telephone prefixes.

Source: own development with usage of Google maps.

III. CLASSIFICATION OF MARKERS WITHOUT TAKING ACCOUNT OF THE SCALE OF THE MAP

The methods described in the previous paragraph are part of the standard Google Maps API development. In its documentation the authors recall the classic methods of cluster analysis by giving them the common name of "*distance based clustering*" but without giving any details of their implementation. To propose such an algorithm first the correct distance definition between markers (representing geographical points) should be introduced.

Let the $r_{earth} = 6371{,}0$ mean the radius of the Earth, $p_1 = (\phi_1, \nu_1)$ $p_2 = (\phi_2, \nu_2)$ – two points on the map described by geographical longitude and latitude. The distance between these points can be defined as:

$$GD(p_1, p_2) = 2r_{earth}\, atgh\left(\frac{\sqrt{\sin(\Delta\phi)^2 \sin(\Delta\nu)^2 \sin(\varphi_1)\cos(\varphi_1)},}{\sqrt{1 - \sin(\Delta\phi)^2 \sin(\Delta\nu)^2 \sin(\varphi_1)\cos(\varphi_1)}} \right) \tag{1}$$

where: $\Delta\phi = \phi_2 - \phi_1$, $\Delta\nu = \nu_2 - \nu_1$.

The proposed algorithm is commonly known *k-means* family descendant and can be stated in four stages (based on [Walesiak, Gatnar ed. 2009]):

a) The starting point is the initial distribution of *s* classes given at random; for each class the centroid is calculated along with GD distances for each object in the cluster.

b) Change assignment of objects into classes with the closest GD distance;

c) Calculate new centroids for each cluster;

d) Repeat steps b) and c) until there will be no moving objects between classes.

Sample results of the algorithm for the actual data examples are illustrated in fig. 3.

IV. CLASSIFICATION OF MARKERS WITH TAKING ACCOUNT OF THE SCALE OF THE MAP

The specificity of the classification of markers on the map is that with a change of scale and display area of the map cluster structure evolves. For example, the markers corresponding to the customers of the company on a map of whole country can be grouped around the major cities and industrial centers forming a focus for these cities. But when zoomed in maps and limited to one city, markers that correspond to the companies can focus around the districts and one large focus is naturally divided into a few smaller ones.

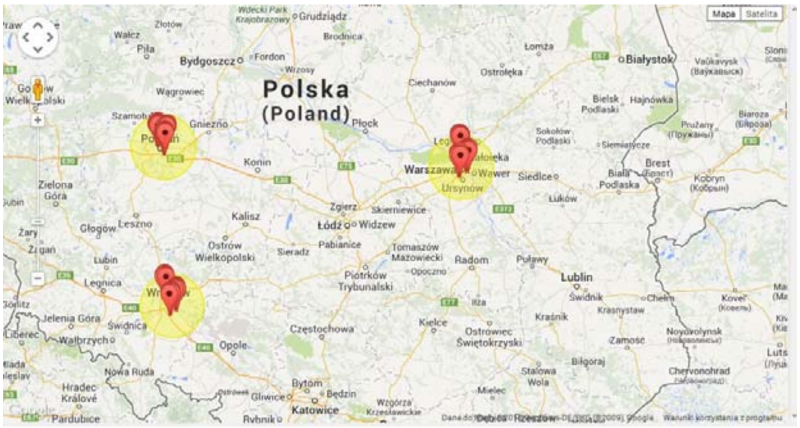


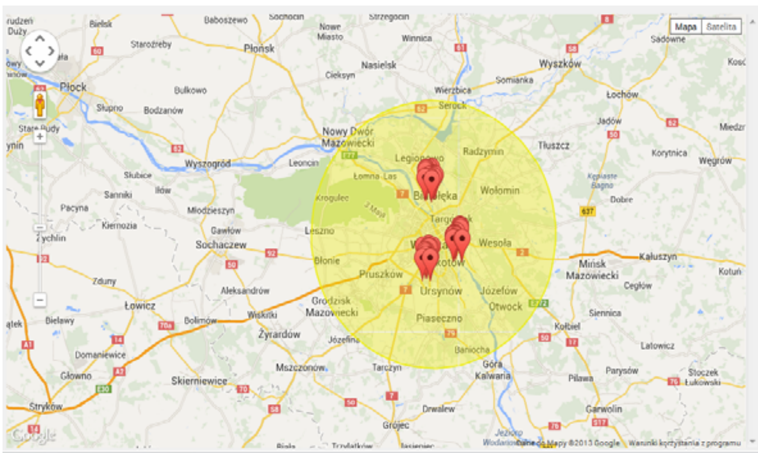
Figure 3. Partitions obtained by the proposed method

Source: own development with usage of Google maps and JavaScript own routines.

There are at least two strategies for correct classification of markers taking account of the scale of the map.

The first is to repeat classification procedure each the map scale is changed or map center is moved. The effect of this approach is shown in fig. 4. Fig. 4 a) shows an invalid cluster structure, duplicated from Fig. 3 and fig. 4 b) contains the correct cluster structure resulting from the repeated cluster analysis procedure. This approach can give appropriate results, but can also drastically slow the process of map display and navigation.

a)



b

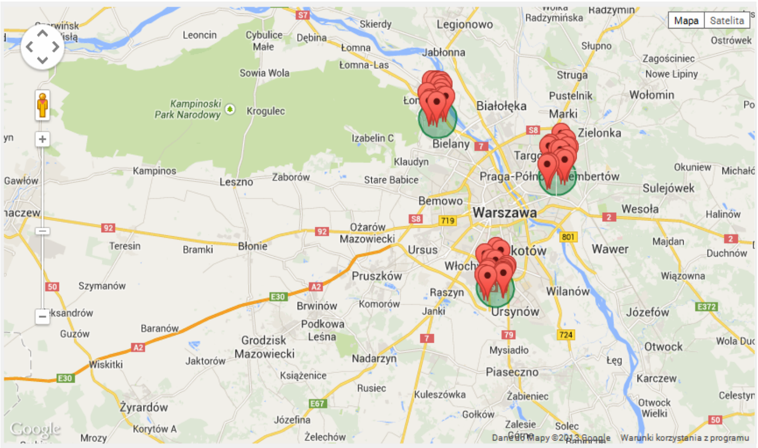


Figure 4. Resulting clusters vs the scale of the map.
a)-data from Figure 3-in an enlarged scale-invalid structure-on cluster.
b)-data from Figure 3-in an enlarged scale, correct structure after the reclassification procedure.

Source: own development with usage of Google maps and JavaScript own routines.

The second approach performs one dynamic classification which would not result in one structure of the clusters, but the hierarchy of clusters giving different results depending on the cut-off level (note that this hierarchy differs from hierarchical agglomerative methods like Ward, McQuitty or complete-link algorithms). The proposal of the algorithm of this type is described in the next section of the paper.

V. ALGORITHM WITH MULTIPLE CLUSTERS STRUCTURES FOR MAP SCALE

For the parameter, the scale of the map (*zoom*) changing from 8 to 20:

a) For each *zoom* algorithm starts with no clusters structure.

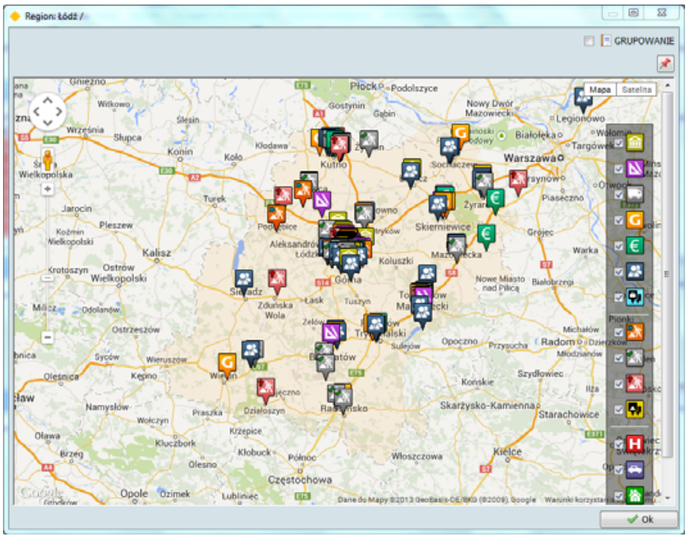
b) Each marker is assigned to nearest cluster that between its centroid and marker is closer than: $d^*(zoom) = \theta_0 2^{zoom_0 - zoom}; zoom_0 = 14; \theta_0 = 1$, if all distances to centroids are greater than $d^*(zoom)$, then new cluster is created.

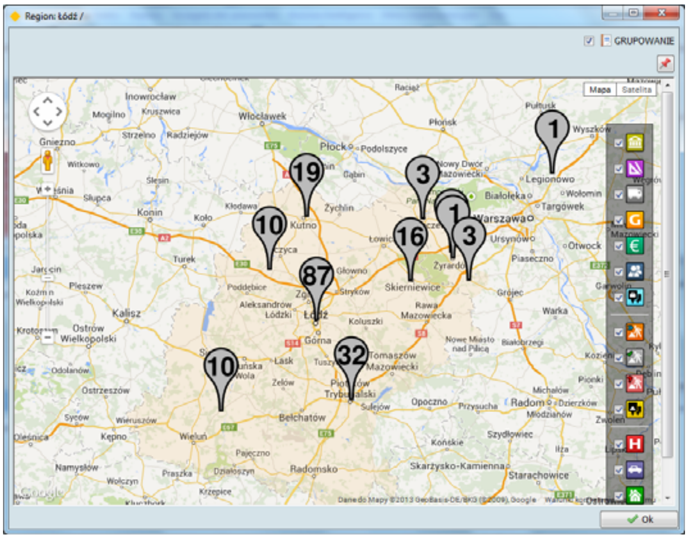
c) Calculate new centroids for each cluster;

d) Repeat steps b) and c) until there are no moving objects between clusters;

e) Remove clusters with no object assigned.

As a result we get the clusters structures for all possible values of the scale of the map. In algorithm we use certain parameters values that should be clarified. Map start scale value equals 8 is adequate to displaying Poland in one window. Map scale equals 20 is the highest possible value of zoom. For map scale equal to 14 ($zoom_0$) one kilometer is equal to nearly 300 pixels on computer screen. Sample results of algorithm on real Lower-Silesia company customers are presented on fig. 5. and 6.





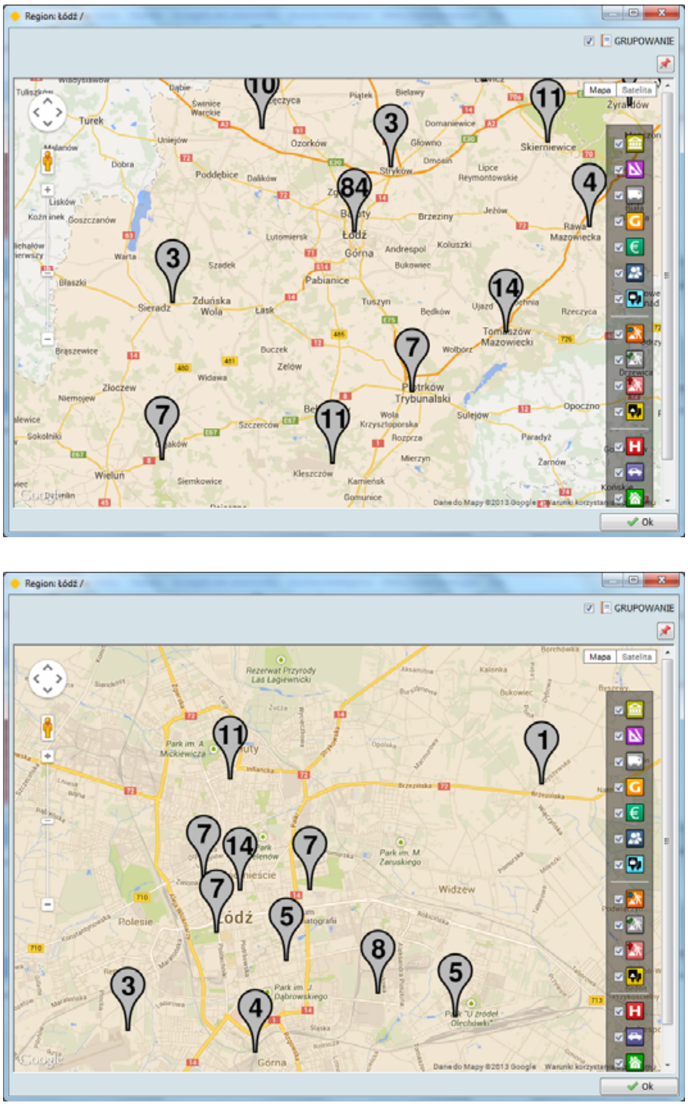


Figure 5. Results of clustering with given algorithm for Łódzkie voivodeship
Source: own development with usage of Google maps and JavaScript own routines.

Fig. 6. Results of clustering with given algorithm for Lubelskie voivodeship

Source: own development with usage of Google maps and JavaScript own routines.

VI. FINAL REMARKS AND OPEN PROBLEMS

In the article dynamical algorithm of geographical points on Google maps that gives different clusters structure for different scale of the map is proposed. Presented algorithm has been implemented in Java/JavaScript and integrated with the Google Maps API.

Two open issues can be stated: The current version of the algorithm does not select an optimal number of clusters (next version could be combined for example with the silhouette index) and there is no objective tool to test methods of classification of geographical points, (such as Rand index). Only visual assessment has been used, but for future versions and for comparison of different algorithms or performance of the same algorithm with different parameters, some kind of benchmark framework should be developed.

REFERENCES

Dudek A. (2011) (red.), *Systemy Informatyczne Zarządzania. Microsoft Business Solutions Navision*, Wydawnictwo UE we Wrocławiu, 2011.

Everitt B.S., Landau S., Leese M. (2001), *Cluster analysis*, Edward Arnold, London.

Gordon A.D. (1999), *Classification*, Chapman & Hall/CRC, London.

Hubert L.J., Arabie P. (1985), *Comparing partitions*, „Journal of Classification", no. 2, 193–218.

Walesiak M., Gatnar E. (red.) (2009), *Statystyczna analiza danych z wykorzystaniem programu R*, PWN, Warszawa.

Too Many Markers! Google Maps API, https://developers.google.com/maps/articles/toomanymarkers#gridbasedclustering

Andrzej Dudek

DYNAMICZNA KLASYFIKACJA PUNKTÓW GEOGRAFICZNYCH NA MAPACH GOOGLE

Klasyfikacja punktów geograficznych na mapach Google jest ciekawym przykładem zastosowań algorytmów analizy skupień, w którym ostateczna liczba otrzymanych skupień jest wynikową nie tylko założeń wstępnych i zastosowanego algorytmu, ale również skali, w której aktualnie jest wyświetlana mapa. Ostatecznym celem klasyfikacji nie jest wyłącznie otrzymanie względnie homogenicznych skupień, ale również zapobieganie zjawisku „zlewania się" markerów na mapie W artykule zaproponowano algorytm automatycznie tworzący strukturę hierarchiczną klas (różniącą się jednak od struktur otrzymywanych w wyniku metod aglomeracyjnych), w taki sposób, aby ostateczna klasyfikacja uwzględniała skalę, w jakiej mapa jest wyświetlana wraz z rzeczywistymi przykładami na mapach Google z wykorzystaniem skryptów JavaScript/ JQuery.