Clustering Algorithm in Order to Find Accident Black Spots Identified By GPS Coordinates

Conference Paper · June 2014

DOI: 10.5593/SGEM2014/B21/S8.063

2 authors:

Sandor Szenasi
Óbudai Egyetem
139 PUBLICATIONS **582** CITATIONS
SEE PROFILE

Peter Csiba
J. Selye University
23 PUBLICATIONS **204** CITATIONS
SEE PROFILE

CLUSTERING ALGORITHM IN ORDER TO FIND ACCIDENT BLACK SPOTS IDENTIFIED BY GPS COORDIANTES

Dr. Sándor Szénási[1]

Dr. Peter Csiba[2]

[1] szenasi.sandor@nik.uni-obuda.hu, Óbuda University - **Hungary**
[2] csiba.peter@selyeuni.sk, J. Selye University - **Slovakia**

ABSTRACT

Black spots are areas in the public road network where the number of accidents is significantly higher than expected. There are several methods that may be used to find these black spots, they usually use the road number and road section based positioning system, which is very useful in cases when we want to find black spots in one road only. But it has several disadvantages in the case of junctions, interurban areas, etc. It would better to use GPS coordinates, but the already existing black spot search methods are not applicable in this case. This paper presents a modified DBSCAN based clustering method in order to find accident black spots. DBSCAN is a general clustering algorithm, so it has been adapted to work with road accidents identified by GPS coordinates. We also present a fast and accurate technique, which calculates the accident density in a given area.

Keywords: DBSCAN, clustering, accident, black spot, algorithm

INTRODUCTION

There are areas in the public road network where the number of accidents is significantly higher than expected, which indicates that their design should be revised. In this paper, we focus especially on personal injury accident accumulation areas (PIAAA). There are several methods [1, 2, 3, 4] that may be used to find these black spots; they are usually based on thresholds or statistical values. For example, the so-called "Marion" method bases itself on the idea of thresholds: a number of accidents and a number of casualties are chosen for a certain period and road section length. A PIAAA is therefore defined as a road section, on which established thresholds are exceeded. Another possible way is to use statistical methods; these detect areas in which accident density is high when compared to a reference density (which can be the density of the entire road network, or the average density of the examined road, etc.). Both methods have advantages and disadvantages. We use the latter, because it is rigorous from a statistical standpoint; however, it is more difficult to understand and check the results.

Classical black spot search methods usually use the road number and road section based positioning system (for example, road 6, 12+300 meters). This is very useful in cases when we want to find black spots in one road only. Using this method, calculating the distance between two accidents is easy (and quick); it is just the difference of the section values. Another advantage is that algorithms based on this positioning system

are usually clean and well understood; it is easy to check the results and fine-tune the parameters.

However, it has some disadvantages. The major ones are the following:

- In the case of junctions, the road number based identification can be very misleading. In the case of an intersection of Road A and Road B, some of the accidents are assigned to Road A and some of them to Road B. From the view of black spot identification, these accidents are closely related, but the road number based approach will not classify these as a black spot. There are several junctions of more than two roads (especially roundabouts), which represent an even worse situation.
- In the case of interurban areas, the road numbering is usually missing or incomplete. In these cases, the positioning system is based on the street name and house number. Accurate distance calculation is almost impossible based on house numbers. Furthermore, the problem described previously about junctions appears even more in the case of interurban areas.

Most accidents occurred in junctions and interurban areas; therefore, these disadvantages cause serious problems. One of the available solutions is to change the positioning system from being classically road number/road section based to another being on geographical coordinates. Nowadays, the proliferation of the GPS system makes this available. However, we should reconsider the already existing black spot searching methods to use the new positioning system.

This paper presents a modified DBSCAN [5] based clustering method in order to find accident black spots. DBSCAN is a general clustering algorithm, so it has been adapted to work with road accidents. We also present a fast and accurate technique, which calculates the accident density in a given area.

BLACK SPOT CLUSTERING

Using GPS coordinates raises many issues. The well-known black spot identification algorithms become unusable; this is because these are usually grounded heavily on the section number based positioning. Of course, we can adapt these algorithms to the new positioning system: the 1 dimensional sections can be replaced by 2 dimensional grids, but the result produced through it is less valuable, and it raises a lot of implementation issues (we need a grid with viable cell sizes, according to the viable section lengths of the 1 dimensional implementation). Another disadvantage is that this method can only identify rectangular black spots, leading to inaccurate accident density values.

Fortunately, there are already several developed density-based clustering techniques in the field of data mining [6]. The principle of density-based methods is as follows: the density of elements within a cluster must be significantly higher than between clusters. That is the main idea for how clusters and outliners can be distinguished.

DBSCAN ADAPTATION

One of the most basic density-based methods is the DBSCAN algorithm (Density-Based Clustering of Application with Noise). An advantage of this algorithm is that given its principle of operation, it is capable of recognising clusters of any shape, and can be used in case of noise as well. In this environment, noise refers to accidents that do not belong to any black spot clusters.

The DBSCAN algorithm uses two main parameters: 1) ε is a radius-type value 2) *MinPts* is a limit for accident numbers. Some consequent definitions: 1) *ε-environment*: the space within a radius of ε of an element; 2) *internal element*: if the ε-environment of an element contains at least MinPts number of elements; 3) *directly densely accessibility:* for a given domain of elements, one element is directly densely accessible from other internal elements if it is the first element's ε-environment; 4) *dense reachability:* it is similar as it is directly densely accessible, but for one element is permitted to be accessible from another only through a chain of directly densely accessible elements; 5) *densely connected elements:* if there is an element from which both are densely reachable with the given parameters; 6) d*ensity-based cluster:* a domain of densely connected elements that shows maximum accessibility of density. Our goal is to find sets of accidents in which all elements are densely connected and no further expansion is possible.

First, we have to define the concept of distance between two elements in the database. In the case of accidents identified by GPS coordinates, this is obvious. We use the Euclidean distance between the two coordinates. Based on this, we can actually calculate the "transitive closed domain" of directly dense accessibilities (this is the maximum domain of densely accessible elements from the starting point). The input of the algorithm is a starting point. First, we check all neighbouring points in its ε-environment. If there are any, we choose the most promising (based on the expected accident density. See below). If there are any acceptable points, we extend the cluster with this point. After that, we start a new iteration, the ε-environment of this new element is also investigated, and if we find acceptable points, they are added to the cluster. If we are unable to select new points, the iteration ends.

AREA CALCULATION

The goal of the original DBSCAN clustering is to get the largest set of items, according to the densely connected criteria. In contrast to the case of accident black spots, we would like to see only accidents that strongly belong to one group (Figure 1). To handle this additional criterion, the search should be supplemented to include a minimum density limit (*MinDensity*). As we had to define a concept of distance between two elements in the database, similarly, for the density calculation we have to define a concept of density of accidents (Equation 1).

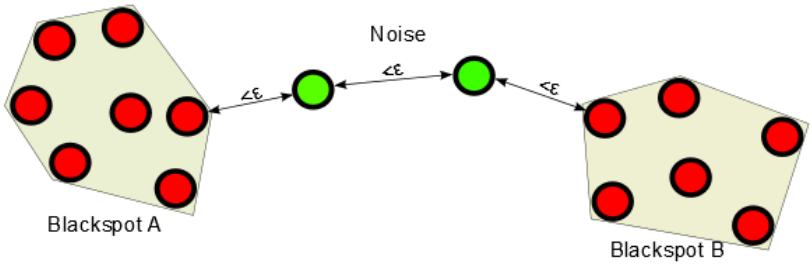

Figure 1 Two independent black spots

$$Cluster\ density = \frac{Accumulated\ weight\ of\ accidents\ in\ the\ cluster}{Area\ of\ cluster} \quad (1)$$

Where the weights of accidents are based on the following:

- Killed participants : 5 point
- Seriously injured participants: 3 point
- Slightly injured participants: 1 point

The area of the cluster is the area of the minimal convex polygon, which contains all the accidents. There are several algorithms to find this polygon for a given point set, but these are usually time consuming. It would better to create a polygon at the start of the DBSCAN algorithm, and maintain it after every expanding operation, based on the following steps:

- Step 1: For clusters with one or two items, this concept cannot be interpreted. This is not an issue, because we usually want to find clusters with more accidents.
- Step 2: For clusters with three items, the area is equal to the area of the triangle spanned by the three points. We need to ensure that the third point is on the right side of the vector connecting the first and second points (we can check this with a single scalar multiplication, and we can reverse the vectors to ensure this constraint).
- Step 3: In the case of every new point (4^{th}, 5^{th},…) we have to check, that the new point is on the right side of all boundary vectors of the actual polygon. If it is true, then the new point is inside the already existing polygon, therefore, we do not have to change it.
- Step 4: If the condition of Step 3 is not true for some of the vectors, then (because of the convexity of the polygon) these must be consecutive vectors, like $P_0 – P_1$ - ... – $P_{k-1} – P_k$. We have to remove these vectors, and add ($P_0 – P_{new}$) and ($P_{new} – P_k$) vectors.

To find the area of the boundary, we can use the shoelace formula, which is a mathematical algorithm that determines the area of a simple polygon whose vertices are described by ordered pairs in the plane.

$$Area = \frac{1}{2}\left|\sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n\right| \quad (2)$$

Where x_i and y_i are the coordinates of the edges.

To optimize this calculation, we can use the following (according to the steps previously mentioned).

- Step 1: We cannot interpret the concept of area.
- Step 2: We apply the shoelace formula for the given three points.
- Step 3: The area of the polygon remains the same.
- Step 4: The area of the polygon: Previous area – removed area + newly added area. Removed and newly added areas are calculated by the shoelace formula based on the removed and added vertexes.

SCORING AND SORTING

By using equation 2, we can calculate the density of an accident black spot. Our goal is to find black spots with maximal density. For this, we start the DBSCAN algorithm from every accident point and evaluate the results (the order can be random [7]).

It is important to note that launching this recursion from every accident shall produce several overlapping clusters (for example, started growings from the accidents of a given black spot). Therefore, an additional procedure is required to eliminate this redundancy. For this, we sort the clusters by accident density, and remove all of them where another overlapping cluster exists with lower index (higher accident density).

Finally we should find the most interesting areas (for this, we can use some fuzzy based methods [8,9]), store them into a tagged database [10] and develop the necessary further steps. Identification of the black spots is only one step of the entire workflow [11,12].

EVALUATION OF RESULTS

The black spot searching algorithm was implemented as a module of a general accident prevention knowledge database. The user should specify the workspace out of the total accident database, using different filtering methods (time, county, road number, etc.), the already mentioned DBSCAN parameters (ε, MinPts, MinDensity, weight of accidents), and some additional parameters (for example, the algorithm can use the traffic database for weighting accidents).

Thanks to the optimization of area calculation (and some geographical indexing), the processing time is acceptable; it takes about 20-30 seconds to process 5000 accidents (this is the number of accidents of an average Hungarian county for 10 years).

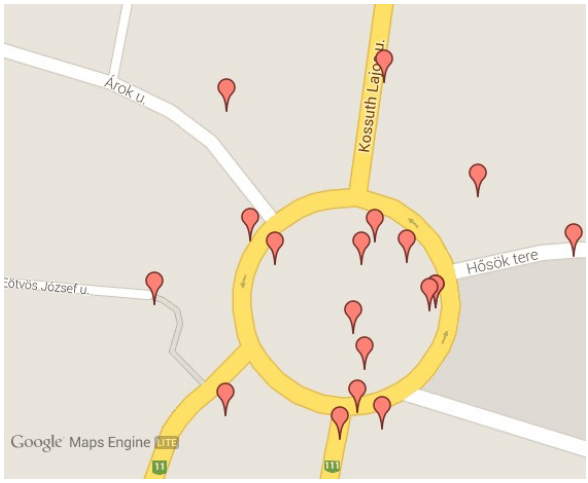

Figure 2 Result of the algorithm (black spot in Esztergom, Hungary)

Figure 2 shows an example, and **Error! Not a valid bookmark self-reference.** shows the detailed data of the given black spot. This black spot is in interurban area and it is in a junction; as it is visible, the new method can find these special areas too.

Table 1 Detailed data of an accident black spot

Road number	Position	GPS latitude	GPS longitude	Street name
11	66+515	47.785717	18.740596	Hősök
11	66+515	47.785717	18.740596	Hősök
11	66+450	47.786110	18.740461	Kossuth Lajos
11	66+523	47.785711	18.740577	Hősök
11	66+506	47.785831	18.740436	Hősök
11	65+925	47.786060	18.740049	Kossuth Lajos
11	66+410	47.785671	18.740380	Kossuth Lajos
11	66+576	47.785722	18.739861	Rudnay Sándor
11326	2+030	47.785792	18.740177	Hősök
11	66+420	47.785792	18.740403	Kossuth Lajos
11	66+510	47.785795	18.740519	Hősök
11	66+530	47.785833	18.740111	Táti
11	66+490	47.785609	18.740410	Kossuth Lajos
111	0+000	47.785533	18.740390	Kiss János Altábornagy
11326	2+015	47.785504	18.740455	Árok
11	66+565	47.785528	18.740047	Táti
11	66+396	47.785486	18.740346	Kossuth Lajos
11326	1+850	47.785911	18.740704	Árok

1111	35+600	47.785806	18.740953	Terézia

The drawback of the algorithm used is that it is quite sensitive to parameters specified by the user (ε, MinPts, MinDns, weight of accidents depending on outcome), and in order to set these correctly, users need to have a certain degree of experience and an understanding of how the algorithm works. The direction for further development may be to find the right parameters [13], or to refine the algorithm by taking the above into account.

REFERENCES

[1] Road Safety Manual. PIARC Technical Committee on Road Safety. 2003

[2] I. Lizamol, A. Shibu, M. S. Saran, Evaluation and treatment of accident black spots using Geographic Information System, International Journal of Innovative Research in Science Engineering and Technology, Vol. 2., No. 8, 2013, pp. 3866-3873.

[3] K. Geurts, G. Wets, Black Spot Analysis Methods: Literature Review, Steunpunt Verkeersveiligheid bij Stijgende Mobiliteit, 2003, pp. 1-30.

[4] E. K. Reshma, S. U. Sharif, Prioritization of Accident Black Spots Using GIS, International Journal of Emerging Technology and Advanced Engineering, Vol 2., No. 9, 2012, pp. 117-122.

[5] J. Han, M. Kamber, Data Mining. Concepts and Techniques. Elsevier, 2001

[6] S. Szénási, D. Jankó, Black spot treatment system using a „hunting for irregular pattern" process and a safety knowledge-base, in On safe roads in the XXI. Century, Budapest, 2006

[7] Gy. Györök, M. Tóth, On Random Numbers in Practice and their Generating Principles and Methods, International Symposium on Applied Informatics and Related Areas: AIS 2010, Székesfehérvár, 2010.11.12, pp. 1-6.

[8] E. Tóth-Laufer, M. Takács, I. J. Rudas, Conjunction and Disjunction Operators in Neuro-Fuzzy Risk Calculation Model Simplification, in 13th IEEE International Symposium on Computational Intelligence and Informatics (CINTI), Budapest, 2012.11.20-22, pp. 195-200.

[9] R. E. Precup, S. Preitl S, J. K. Tar, M. L. Tomescu, M. Takács, P. Korondi, P. Baranyi, Fuzzy control system performance enhancement by iterative learning control, in IEEE Transactions on Industrial Electronics, 2008, pp. 3461-3475.

[10] Á. Bogárdi-Mészöly, A. Rövid, H. Ishikawa, S. Yokoyama, Z. Vámossy, Tag and Topic Recommendation Systems, ACTA POLYTECHNICA HUNGARICA, Vol. 10., No. 6, 2013, pp. 171-191.

[11] M. Kozlovszky, K. Karoczkai, I. Marton, A. Balasko, A. Marosi, P. Kacsuk, Enabling Generic Distributed Computing Infrastructure Compability for Workflow Management Systems, COMPUTER SCIENCE, Vol. 13., No. 3., 2012, pp. 61-78.

[12] J. Tick, Cs. Imreh, Z. Kovács, Business Process Modeling and the Robust PNS Problem, ACTA POLYTECHNICA HUNGARICA, Vol. 10, No. 6., 2013, pp. 193-204.

[13] S. Sergyán, L. Csink, Automatic Parameterization of Region Finding Algorithms in Gray Images, in 4th International Symposium on Applied Computational Intelligence and Informatics, Timisoara, 2007.05.17-18., pp. 199-202.