Determining DBSCAN-Entrpopy Hybrid Algorithm Parameters for Converting GPS Points to Activities

true true

2021-11-30

Question

Global Positioning System (GPS) surveys have become a more accurate and reputable alternative to previous travel survey methods that collect activity-travel patterns. Despite GPS devices' ability to record time and positional characteristics, they still require two steps, cleaning and processing, in order to convert the positional characteristics into trip purposes and activities.

Currently, many researchers use time and speed rule-based algorithms to define when and where activities occur (Shen and Stopher 2014). Due to their subjective nature, these rules are not ideal. For example, people walk at different speeds, so the speed threshold at which someone is considered stagnant would require manual changing from person to person. If not changed, the number of activities for each person could be misleading and inaccurate. These issues may explain why rule-based algorithms' accuracies typically range from 43% to 61% (Shen and Stopher 2014). While these processing results (Step 2) are not ideal, the data-cleaning method (Step 1) still serves as a reputable guide regardless of the processing method.

Therefore, once the data is cleaned with accordance to rule-based algorithm methods, a DBSCAN and entropy based algorithm should be applied for the processing step. In this type of algorithm, four parameters are needed to determine an activity: minimum number of points (minpts) within a predefined radius (eps) with a minimum amount of entropy (entr_t) (Gong L. 2018), and after a certain period of time from the previous activity (delta_t) . When selected properly, these parameters will not need to change from person to person and therefore often lead to more accurate activity counts. In fact, one experiment (Luo et al. 2017) using just a DBSCAN cluster-based algorithms proved to be 92% precise.

One way to select the minPoints and radius (eps) thresholds is to arbitrarily pick the minPoints based on how large the data set is (with a minimum of three) and then set k = minPts in a k-distance plot (Kassambara 2018). Good values of the radius value is where the k-distance plot shows a strong bend. Another method involves calculating the arithmetic mean and standard deviation of a synthetic GPS trajectory, and subject those values to a Gaussian curve equation to solve for eps given an arbitrary minPts (Xiu-Li and Wei-Xiang 2009). Unfortunately, these methods only work in a pure DBSCAN algorithm where only minpts and eps are accounted for, not entr_t and delta_t.

Hence, the purpose of this paper is to explore a method of how to simultaneously select all four parameters as accurately as possible in a DBSCAN entropy based algorithm after the GPS data has been cleaned.

Methods

Data

The GPS data used to determine the four most accurate parameters come from 60 volunteers in the Utah County area and were taken over a period of six or more months depending on the person. An example of what the raw GPS data looked like is shown in Figure 2.1.

Table 1: (#tab:Figure1)Raw GPS Data

accuracy	timestamp	speed	lat	lon	time
16	2021-03-17 22:59:36	-1	40.25293	-111.6602	1.616044e+12
16	2021-03-17 22:59:37	-1	40.25293	-111.6602	1.616044e+12
16	2021-03-17 22:59:38	-1	40.25293	-111.6602	1.616044e+12
16	2021-03-17 22:59:39	-1	40.25293	-111.6602	1.616044e+12
16	2021-03-17 22:59:40	-1	40.25293	-111.6602	1.616044e+12
16	2021-03-17 22:59:41	-1	40.25293	-111.6602	1.616044e+12

Before the GPS data can be processed, it had to be cleaned and reformatted. Since a DBSCAN algorithm is being used, the speed variable was removed completely. From there, the dates and times had to be reformatted using functions from the **lubridate** package in R and by writing a "Yesterday" function with output "ActivityDay" that defines activity days as being from 3 AM to 3 AM instead of 12 AM to 12 AM. This was done because many respondents are college students, so they are likely to make trips after midnight. Table 2.2 shows what the cleaned data looked like.

Table 2: (#tab:Figure2)Cleaned GPS Data

lat	lon	timestamp	date	time	activityDay
40.25293	-111.6602	2021-03-17 22:59:36	2021-03-17	22:59:36	17-3
40.25293	-111.6602	2021-03-17 22:59:37	2021-03-17	22:59:37	17-3
40.25293	-111.6602	2021-03-17 22:59:38	2021-03-17	22:59:38	17-3
40.25293	-111.6602	2021-03-17 22:59:39	2021-03-17	22:59:39	17-3
40.25293	-111.6602	2021-03-17 22:59:40	2021-03-17	22:59:40	17-3
40.25293	-111.6602	2021-03-17 22:59:41	2021-03-17	22:59:41	17-3

Models

Once the data is cleaned and properly formatted, it is run through a DBSCAN-entropy hybrid algorithm largely based on the method created by Gong et al. in 2018 (Gong L. 2018). After the DBSCAN algorithm determines how many total clusters there are based on the eps and minpts parameters, they get further split based on the delta_t parameter if necessary. If the time difference between points at the same place is greater than delta_t, then the points will be split into two separate clusters or activities.

Finally, there is an entropy calculation step where entropy is determined by the change in departure angle between consecutive points. The equation for this entropy is shown in the equation below (Gong L. 2018). If the points are in a line, the entropy is very low. In this algorithm, the entr_t parameter determines at which entropy someone is actually likely to be moving and not just at a stoplight, etc.

$$EI_q = -\sum_{d=1}^{D}((\frac{n_d}{N})ln(\frac{n_d}{N})) \tag{1}$$

For this experiment, I will look at a map of the unprocessed, cleaned data and determine with my eyes how many clusters there are. Then, the hybrid algorithm will calculate the number of clusters using randomized values for the four parameters. Based on all the previous research that has been discussed, the ranges for the possible parameters are as follows:

minpts: (3,10) eps: (1,50) delta_t: (300, 1500) seconds entr_t: (0.5,3)

Then, I will compare the amount of clusters I saw to the number of clusters the hybrid algorithm calculated. This process was repeated 5 times over 10 different days in order to determine which parameters are the most accurate.

Findings

The hybrid algorithm returned a tibble with the parameters randomly selected for each round. In this case, 5 rounds of parameters were performed for each date. The first 10 rows of the resulting tibble, called random_clusters when one runs "tar_make()", is shown below:

```
## # A tibble: 5 x 7
##      eps minpts delta_t entr_t  draw params    clusters
##    <dbl>  <int>   <dbl>  <dbl> <int> <list>    <list>
## 1 11.0        1    8.08   1.25     1 <dbl [4]> <tibble [4 x 4]>
## 2  0.170      1   70.6    2.02     2 <dbl [4]> <tibble [4 x 4]>
## 3  3.10       1  194.     1.79     3 <dbl [4]> <tibble [4 x 4]>
## 4 28.7        1  850.     1.89     4 <dbl [4]> <tibble [4 x 4]>
## 5 31.0        1   80.8    2.10     5 <dbl [4]> <tibble [4 x 4]>
```

As seen from the random_clusters object, the number of clusters and information about those clusters are stored as tibbles in the column "clusters". An example of the contents of one of those "clusters" tibbles is seen in Figure 1.

```
## [[1]]
## # A tibble: 4 x 4
##   date       data              n         clusters
##   <date>     <list>            <list>    <list>
## 1 2020-09-04 <sf [17,814 x 6]> <int [1]> <sf [148 x 6]>
## 2 2020-09-05 <sf [28,800 x 6]> <int [1]> <sf [254 x 6]>
## 3 2020-09-06 <sf [12,182 x 6]> <int [1]> <sf [31 x 6]>
## 4 2020-09-08 <sf [16,472 x 6]> <int [1]> <sf [159 x 6]>
```

Figure 1 also shows the 10 dates that were analyzed for this report as well as the associated nested geometric GPS locations (data) and nested number of clusters (clusters). Further expanding the clusters column list in Figure 1 shows the number of clusters for that date using the parameters from the corresponding iteration. In other words, each date had its number of clusters calculated using 5 different sets of parameters.

Finally, those number of clusters (activities) were all compared to the number of clusters seen by looking at a map of the raw GPS points created using the *leaflet* and *sf* packages in R (example shown in Figure 2). The error between the algorithm's calculated clusters for each set of parameters and the clusters seen from the maps was calculated. Whichever set of parameters consistently gave the lowest error for each date is decidedly the most accurate set of parameters to use for this DBSCAN entropy hybrid algorithm.

```
## [[1]]
```

Based on the results shown in Table 2, the parameters eps = , minpts = , delta_t, and entr_t = consistently gave the smallest errors across all 10 days. Therefore, that set of four parameters together are the most accurate to use in this kind of DBSCAN Entropy hybrid method where all four are being used simultaneously, impacting each other, in order to convert cellular GPS data into activities. An important note is that this is based on only performing 5 random iterations of parameters samples. In theory, there are many more combinations of all four of these parameters within their given ranges, so more testing could provide even more accurate results.

Table 3: (#tab:exampleError)Algorithm vs. Manual Error

eps	minpts	delta_t	entr_t	draw	date	alg_clusters	man_clusters	error
11.00	1	8.08	1.25	1	9/4/2020			
0.17	1	70.60	2.02	2	9/4/2020			
3.10	1	194.00	1.79	3	9/4/2020			
28.70	1	850.00	1.89	4	9/4/2020			
31.00	1	80.80	2.10	5	9/4/2020			
11.00	1	8.08	1.25	1	9/5/2020			

Acknowledgements

Gong L., Morikawa T, Yamamoto T. 2018. "Identification of Activity Stop Locations in GPS Trajectories by DBSCAN-TE Method Combined with Support Vector Machines." *Transportation Research Procedia*. https://doi.org/10.1016/J.TRPRO.2018.10.028.

Kassambara, Alboukadel. 2018. "DBSCAN: Density-Based Clustering Essentials." *DataNovia*. https://www.datanovia.com/en/lessons/dbscan-density-based-clustering-essentials/.

Luo, Ting, Xinwei Zheng, Guangluan Xu, Kun Fu, and Wenjuan Ren. 2017. "An Improved DBSCAN Algorithm to Detect Stops in Individual Trajectories." *ISPRS International Journal of Geo-Information* 6 (3). https://doi.org/10.3390/ijgi6030063.

Shen, Li, and Peter R. Stopher. 2014. "Review of GPS Travel Survey and GPS Data-Processing Methods." *Transport Reviews* 34 (3): 316–34. https://doi.org/10.1080/01441647.2014.903530.

Xiu-Li, Zhao, and Xu Wei-Xiang. 2009. "A Clustering-Based Approach for Discovering Interesting Places in a Single Trajectory" 3: 429–32. https://doi.org/10.1109/ICICTA.2009.569.