My Maps

Gillian Riches[a,*], Gregory Macfarlane[a,**]

[a] *Civil and Environmental Engineering Department, 232 Engineering Building, Provo, Utah 84602*

Abstract

This is where the abstract should go.

Keywords: GPS Data, Trips, Clusters

1. 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 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 results are not ideal, the initial data cleaning process these methods undergo prior to processing is a useful reference.

Once the data is cleaned and ready to be analyzed, a cluster-based algorithm should be applied instead of a rule-based algorithm. In a cluster-based algorithm, the density of GPS points within a predefined radius determines an activity. Although the radius and point density values are still parameters that the researcher needs to choose in the beginning, they would not vary from person to person. Therefore, when selected properly, these objective parameters lead to more accurate activity counts. In fact, one experiment (Luo et al., 2017) using a DBSCAN cluster-based algorithms proved to be 92% precise.

One way to determine 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) .

The purpose of this paper is to determine how much the minPts threshold affects the eps value and, subsequently, which DBSCAN minPts and eps parameters would work best for my particular GPS data set after it has been cleaned.

[*] Corresponding Author

[**] Present affiliation: Committee Chair

Email addresses: `martingillian4@gmail.com` (Gillian Riches), `gregmacfarlane@byu.edu` (Gregory Macfarlane)

2. Methods

2.1. Data

The GPS data used to determine the minPts and eps 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: 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 that defines days as being from 3 AM to 3 AM instead of 12 AM to 12 AM. This was done because large amount of the demographic is college students, and they are likely to make trips after midnight. Table 2.2 shows what the cleaned data looked like.

Table 2: Cleaned GPS Data

id	lat	lon	timestamp	date	time
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:36	2021-03-17	22:59:36
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:37	2021-03-17	22:59:37
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:38	2021-03-17	22:59:38
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:39	2021-03-17	22:59:39
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:40	2021-03-17	22:59:40
5f5184e73e2fd848eac22aec	40.25293	-111.6602	2021-03-17 22:59:41	2021-03-17	22:59:41

2.2. Models

Once the data is cleaned and properly formatted, it is run through a DBSCAN algorithm largely based on the method done by Gong et al. in 2018 (Gong L., 2018). This is where the eps and MinPts parameters will be chosen. After the DBSCAN algorithm determines how many clusters there are, they get split based on the time between consecutive points. This is to remove any error that comes from when a person makes two trips to the same place in one day (Ex. Home). Now, that cluster counts as two separate clusters if the time difference between the clusters is large enough.

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 person is likely at a stoplight and the entropy is very low. Implementing this entropy equation also removes the need for any subjective time or speed rules. For the purposes of this question, the entropy constraint will always be set to 0.5. The only changing parameters will be MinPts and eps.

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

The resulting clusters from this DBSCAN and entropy hybrid model are displayed in maps using R. For this experiment, I will look at a map of the cleaned data and determine how many clusters it looks like there is. Then, the data goes through the model and the calculated number of clusters is compared to how many I saw originally. I repeated this 5 times over 10 days in order to determine which eps and MinPts have the lowest error.

3. Findings

This section might be called "Results" instead of "Applications," depending on what it is that you are working on. But you'll probably say something like "The initial model estimation results are given in Table **??**." That table is created with the `modelsummary()` package and function.

With those results presented, you can go into a discussion of what they mean. first, discuss the actual results that are shown in the table, and then any interesting or unintuitive observations.

3.1. Additional Analysis

Usually, it is good to use your model for something.

- Hypothetical policy analysis
- Statistical validation effort
- Equity or impact analysis

If the analysis is substantial, it might become its own top-level section.

Acknowledgements

This is where you will put your acknowledgments

References

Gong L., Yamamoto T, M. T. (2018). Identification of activity stop locations in gps trajectories by dbscan-te method combined with support vector machines. *Transportation Research Procedia.*

Kassambara, A. (2018). Dbscan: Density-based clustering essentials.

Luo, T., Zheng, X., Xu, G., Fu, K., and Ren, W. (2017). An improved dbscan algorithm to detect stops in individual trajectories. *ISPRS International Journal of Geo-Information*, 6(3).

Shen, L. and Stopher, P. R. (2014). Review of gps travel survey and gps data-processing methods. *Transport Reviews*, 34(3):316–334.

Xiu-Li, Z. and Wei-Xiang, X. (2009). A clustering-based approach for discovering interesting places in a single trajectory. 3:429–432.