Relationships of Compression Ratio and Error in Trajectory Simplification Algorithms

Gary Reyes[1](✉), Víctor Maquilón[1], and Vivian Estrada[2]

[1] Facultad de Ciencias Matemáticas y Físicas, Universidad de Guayaquil, Cdla. Universitaria Salvador Allende, Guayaquil 090514, Ecuador
{gary.reyesz,victor.maquilonc}@ug.edu.ec

[2] Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños km 2 1/2, La Habana, Cuba
vivian@uci.cu

Abstract. GPS trajectory simplification algorithms are of great importance for GPS data analysis and processing. The correct selection of these algorithms in accordance with the type of trajectory to be analyzed facilitates the reduction of storage and processing space in data analysis. This paper analyzes the correlation between the compression ratio of GPS trajectory simplification algorithms and their margin of error. These metrics measure the effectiveness of simplification algorithms in general and this work focuses specifically on batch simplification. For this purpose, coordinates in GPS trajectories of different sets of data are used and the algorithms are executed on them, taking into account that the analysis performed for the simplification process takes into account the beginning and the end of each trajectory. Finally, the data obtained from the experiments are presented in tables and figures. The experiments show that TD-TR has better performance than Douglas-Peucker algorithm due to the selected variables.

Keywords: Trajectory simplification · GPS trajectories · Compression ratio

1 Introduction

A trajectory is represented as a discrete sequence of geographic coordinate points [1]. An example of trajectories are vehicular trajectories that are composed of thousands of points, since the stretches traveled in cities are usually long and with many stops, which implies a greater emission of coordinates generated from GPS devices. There are currently active research areas related to GPS trajectories. Among these is the area of trajectory pre-processing, which studies trajectory simplification techniques and algorithms. The trajectory simplification algorithms eliminate some sub traces of the original route [2]; which decreases the data storage space and data transfer time. A framework where these areas are observed is proposed in [3].

Reducing the data size of a trajectory facilitates the acceleration of the information extraction process [4]. There are several path simplification methods that are suitable for different types of data and yield different results; but they all have the same principle

R. Valencia-García et al. (Eds.): CITI 2021, CCIS 1460, pp. 140–155, 2021.
https://doi.org/10.1007/978-3-030-88262-4_10

in common: compressing the data by removing the redundancy of the data in the source file [5–7]. GPS trajectory simplification algorithms can be classified into [8]:

- Online algorithms
- Batch algorithms

Online algorithms do not need to have the entire trajectory ready before starting simplification, and are suitable for compressing trajectories in mobile device sensors [9, 10]. Online algorithms not only have good compression ratios and deterministic error bounds, but also are easy to implement. They are widely used in practice, even for freely moving objects without the constraint of road networks [9, 11]. Batch algorithms require all points in the trajectory before starting the simplification. Batch algorithms have all the points in the trajectory before starting the simplification process, which allows them to perform better processing and analysis of these [12]. Among the most commonly used batch GPS trajectory simplification algorithms according to the analyzed literature [13–16] are:

- Douglas-Peucker algorithm.
- TD-TR algorithm.

From the documentary analysis performed, a set of common deficiencies were identified in the aforementioned batch GPS trajectory simplification algorithms [17]. These deficiencies attempt against the efficiency of batch simplification algorithms, the efficiency in these algorithms is given by the compression ratio and the error (margin of error).

This work is focused in the GPS trajectory simplification algorithms to address the problem of storage space optimization in order to face of the large volumes of data that are generated today from GPS trajectories. For this purpose, we analyze the compression levels originated by the use of simplification algorithms, but it is also important to analyze the relationship between these compression levels and the quality of the resulting trajectory denoted by the margin of error. The values obtained from the correlation of these two variables are analyzed according to the elimination of points considered non-significant. The analysis of the results allows the use of the appropriate algorithms according to their compression levels and the data characteristics to be processed.

The present work identifies the relation between the compression ratio and the margin of error, and is organized as follows: in the previous work session, a description of the algorithms to be experimented is given. In the experimentation session, the experiments performed and the data sets used are described. In the discussion session, the results obtained are described briefly and concisely.

2 Previous Work

GPS trajectory simplification is an active field of research, especially in recent years with the increasing use of this technology [18]. In the present work, it is considered important to analyze the Douglas-Peucker and TD-TR algorithms in order to compare their performance in terms of the selected metrics.

2.1 Douglas-Peucker Algorithm

Douglas-Peucker (DP) is a GPS trajectory simplification algorithm that uses the top-down method for data analysis. It is used to remove a series of line segments from a curve, which reduces the amount of data present in a GPS trajectory [19]. The selection of points is performed recursively and on the original series [9, 20–22].

Douglas-Peucker implements a divide and conquer strategy and is composed of several steps which are listed below [23, 24]:

1. The first and last nodes of a polygonal chain are taken as endpoints of a line.
2. For all intermediate points, the distances less than this line are determined. If it is greater than any distance, the polygonal chain is separated with the point having the greater distance, forming two new segments.
3. The new segments formed are analyzed by performing the same procedure.
4. It ends when it does not exceed any point line distance.

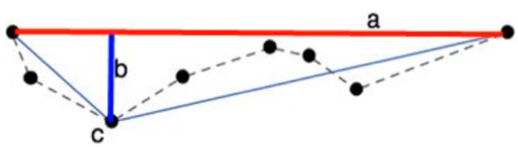

Fig. 1. Douglas-Peucker algorithm

As shown in Fig. 1 the algorithm starts with an initial data set formed by n points, where we start from the initial point (x_1, y_1) drawing a straight line towards the final point of the data set (x_n, y_n), in the graph it is represented with the letter a.

Obtaining the straight line we proceed to take the following points (x_2, y_2), (x_3, y_3).....(x_n, y_n) and it is verified if its perpendicular distance (letter b) with respect to the line is greater than the tolerance entered, once the farthest point is obtained, this is taken as a reference to repeat the process, that is, straight lines are drawn from (x_1, y_1) to the farthest point found (x_3, y_3) and from (x_3, y_3) to (x_n, y_n), repeating the process of obtaining the point with the tolerance greater than the one entered.

2.2 TD-TR Algorithm

This algorithm is a modification of the Douglas-Peucker algorithm where the time variable is added. For this, the coordinates of point P_i' in time are calculated by the ratio of two time intervals. The difference between the time of the end point and the time of the initial point is calculated using the formula 1

$$\Delta_e = t_e - t_s. \tag{1}$$

The difference between the time of the point to be analyzed and the time of the starting point is calculated using the formula 2:

$$\Delta_i = t_i - t_s \tag{2}$$

To obtain the coordinates of P_i' formulas 3 and 4 are applied:

$$x_i' = x_s + \frac{\Delta_i}{\Delta_e}(x_e - x_s) \tag{3}$$

$$y_i' = y_s + \frac{\Delta_i}{\Delta_e}(y_e - y_s) \tag{4}$$

After obtaining the coordinates, the synchronous Euclidean distance between P_i' and P_i, is calculated. If the distance is greater than the tolerance, that reference point is taken and the calculation of the intervals is performed again. The worst-case computational complexity of TD-TR is O(n^2), since it extends the original Douglas-Peucker algorithm. The improved implementation of O($n\log_n$) for Douglas-Peucker that takes advantage of geometric properties cannot be applied to TD-TR [25]. In Fig. 2 the TD-TR algorithm is shown.

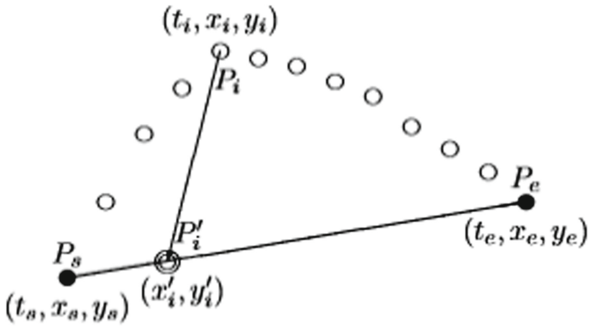

Fig. 2. Process of trajectory simplification using TD-TR algorithm

3 Methodology

The methodology applied in this work is composed by the following steps:

- Literature review and identification of main characteristics of the algorithms to be analyzed.
- Implementation of the Douglas Peucker and TD-TR algorithms in R language.
- Adaptation for the algorithms to process the three Datasets used in the experimentation.
- Implementation of the calculations of the compression ratio and error metrics to the results of the execution of the algorithms.
- Generation of log files for analysis and application of descriptive statistics measures to the compression ratio and error metrics.
- Application of a Correlation Test to the compression ratio and error metrics.

It is important to mention that part of the methodology is the final visualization of the processed trajectories before and after simplification, which for the purpose of this work was not necessary.

4 Experimentation and Results

This session describes the experiments carried out and presents the results achieved for the compression ratio and margin of error metrics with both the Douglas Peucker algorithm and the TD-TR algorithm.

The experimentation process was carried out by implementing the Douglas Peucker and Top Down Time Ratio (TD-TR) algorithms in R language. For the execution of the experiments, a laptop PC with an Intel core i5-8250 CPU @ 1.60 GHz, 8 GB RAM memory and 64bit architecture was used. As data sets were used:

- Guayaquil city dataset[1]
- Beijing city dataset[2]
- Aracaju city dataset.[3]

The Guayaquil-Ecuador trajectories dataset corresponding to the trips made by 218 university students in some means of transportation such as cab, motorcycle, metro-via, collected in the city of Guayaquil, Ecuador. It contains GPS coordinates collected in the northern and central sectors of Guayaquil during October 28, 2017. The locations in this dataset were collected by smartphones with an average time interval between two consecutive locations of 5 s. The file format for each mobility trajectory is as follows: Each record contains id trajectory, latitude, longitude, time, username, email, and time. The data were pre-processed and after filtering by time 30557 records were obtained representing 206 trajectories from the entire dataset.

The Beijing dataset was collected from cab routes in the city of Beijing, China. The dataset comes from the t-drive project and contains GPS coordinates of 10,357 cabs collected in Beijing during February 02, 2008. The locations in this dataset were collected by GPS devices from taxis with average devices with an average time interval between two consecutive locations of 177 s. The file format of each trajectory is as follows: Each record contains. id trajectory, date-time, longitude, latitude. The data were pre-processed and after filtering by time 62138 records were obtained representing 630 trajectories from the whole dataset.

The Aracaju dataset used within the experiments is formed by user paths in cars or buses through the Go! Track application in the city of Aracaju, Brazil. The dataset comes from trajectories collected in Aracaju during September 13, 2014 and January 19, 2016. The locations in this dataset were collected with high time variability between two consecutive locations.

The file format for each trajectory contains a series of records. Each record has id, latitude, longitude, track id, time. The data were pre-processed and filtered by time. In this database are 14096 records which represent 137 trajectories.

The formulas for the compression ratio and error indicators were implemented so that they can be calculated once each of the algorithms processes different samples from the three data sets. For this purpose, the sample size calculation for finite populations

[1] https://github.com/gary-reyes-zambrano/Guayaquil-DataSet.git.
[2] https://www.microsoft.com/en-us/research/publication/tdrive-trajectory-data-sample.
[3] https://archive.ics.uci.edu/ml/machine-learning-databases/00354/.

was performed. The sampling method used was simple random. Compression ratio: It is defined as the size of the original trajectory divided by the size of the compressed representation of that trajectory. For example, a compression ratio of 50 indicates that only 2% of the original points remain in the compressed representation of the trajectory [4]. Formula 5 is used to calculate the compression ratio where n represents the total number of points and x represents the number of resulting points after applying the corresponding compression technique.

$$r(x) = \frac{n - x}{n} * 100 \tag{5}$$

Margin of Error: Where SED (Synchronized Euclidean Distance) is defined in Equation 2.1 as the synchronous Euclidean distance and the distortion units between a point on the original trajectory and its corresponding synchronous point on the approximate trajectory are calculated [26]. The maximum of the distances is selected to evaluate the point acceptance condition.

$$SED(pti, pti) = \sqrt{\left(x_{ti} - x'_{ti}\right)^2 + \left(y_{ti} - y'_{ti}\right)^2} \tag{6}$$

To evaluate the distortion of the whole trajectory, the maximum synchronized Euclidean distance is used, which is defined as:

$$SED(P, P') = MAX\,(SED(pti, pti')) \tag{7}$$

4.1 Douglas Peucker Algorithm Evaluation Result

The descriptive statistics measured for the compression ratio and error variables are shown in Table 1.

Table 1. Descriptive Statistics for Compression Ratio and Margin of Error (SED) - Douglas Peucker Algorithm

	Aracaju		Beijing		Guayaquil	
	Compression ratio	SED	Compression ratio	SED	Compression ratio	SED
Valid	101	101	230	230	134	134
Missing	0	0	0	0	0	0
Mean	75.085	0.746	62.412	0.419	89.107	0.787
Std deviation	31.559	1.349	19.885	0.804	15.462	0.964
Skewness	−1.721	3.208	−0.927	6.370	−4.766	2.508
Std. Error of Skewness	0.240	0.240	0.160	0.160	0.209	0.209

(*continued*)

Table 1. (*continued*)

	Aracaju		Beijing		Guayaquil	
	Compression ratio	SED	Compression ratio	SED	Compression ratio	SED
Kurtosis	1.486	12.697	0.976	55.367	24.391	7.121
Std. Error of Kurtosis	0.476	0.476	0.320	0.320	0.416	0.416
Range	99.342	8.070	99.558	8.752	98.802	5.454
Minimum	0.000	0.000	0.000	0.000	0.000	0.000
Maximum	99.342	8.070	99.558	8.752	98.802	5.454

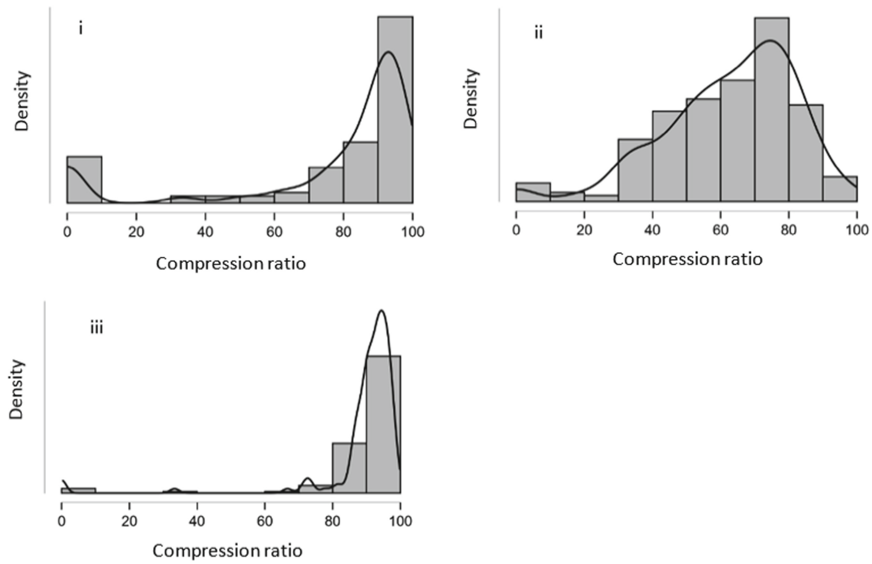

Fig. 3. Compression Ratio Distribution in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Douglas Peucker Algorithm

The compression ratio distribution plot evidences negative skewness (Skewness < 0) with leptokurtic kurtosis (Kurtosis > 0). The curves for the three data sets are shown in Fig. 3.

The SED plot shows a positive skewness (Skewness > 0) with leptokurtic kurtosis (Kurtosis > 0). The curves for the three sets of data are shown in Fig. 4.

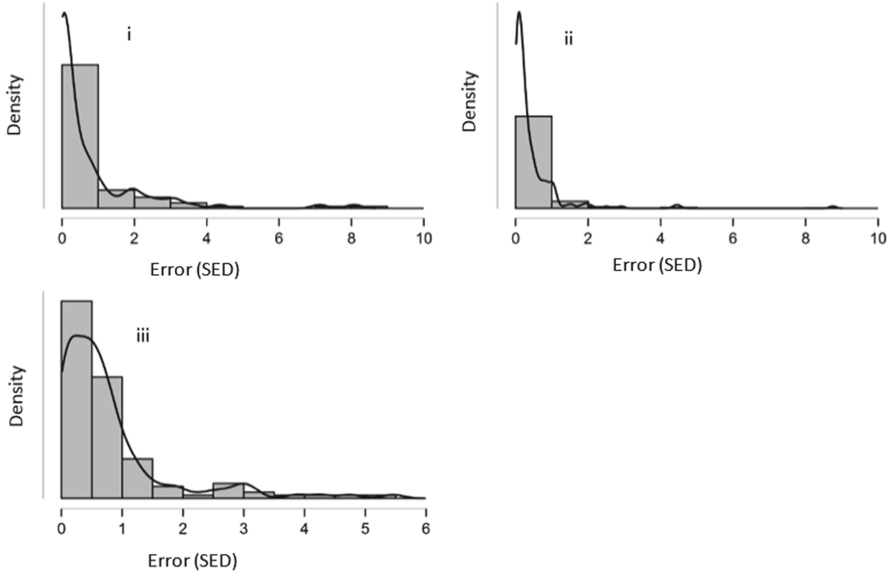

Fig. 4. Error Distribution (SED) (margin error) in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Douglas Peucker Algorithm.

Pearson's correlation test between compression ratio and margin error (SED) for the Aracaju dataset, shows a significantly positive correlation with r = 0.313 and p-value = 0.001, as shown in Table 2.

Table 2. Pearson's Correlation for Compression Ratio and Error (SED) in the Aracaju data set - Douglas Peucker Algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.313	n/a
	p-value	0.001	n/a

Pearson's correlation test between compression ratio and error margin for the Beijing data set evidences a significantly positive correlation with r = 0.376 and p-value < 0.001, as shown in Table 3.

Pearson's correlation test between the compression ratio and the margin error (SED) for the Guayaquil data set shows a significantly positive correlation with r = 0.248 and p-value = 0.004, as shown in the Table 4.

A visual inspection of the scatter plot, showed in Fig. 5, for the three data sets shows the correlation between compression ratio and margin of error.

Table 3. Pearson's Correlation for Compression Ratio and Error (SED) on Beijing data set - Douglas Peucker Algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.376	n/a
	p-value	<.001	n/a

Table 4. Pearson's Correlation for Compression Ratio and Error (SED) on Guayaquil data set - Douglas Peucker Algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.248	n/a
	p-value	0.004	n/a

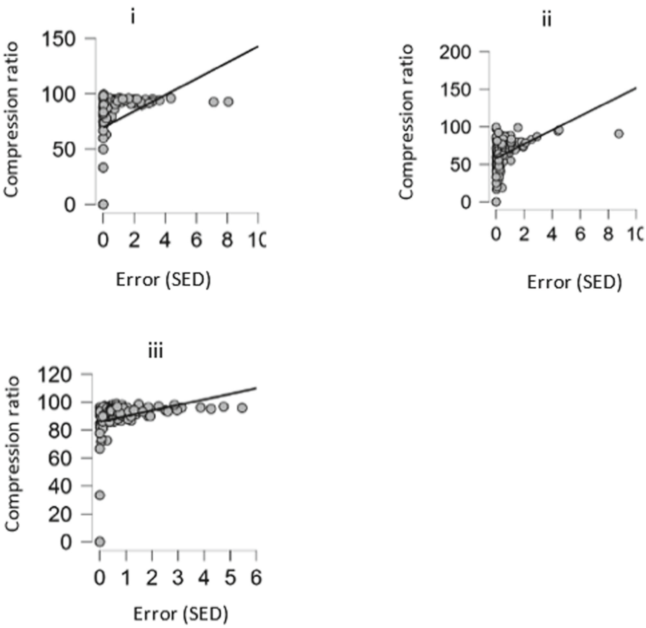

Fig. 5. Dispersion of the Correlation between Compression Ratio and Error (SED) (margin error) in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Douglas Peucker Algorithm.

4.2 Top Down – Time Ratio (TD-TR) Algorithm Results

The descriptive statistics measured for the compression ratio and margin error variables are shown in Table 5.

Table 5. Descriptive Statistics for the Compression Ratio and Error (SED) - TD-TR Algorithm

	Aracaju		Beijing		Guayaquil	
	Compression ratio	SED	Compression ratio	SED	Compression ratio	SED
Valid	101	101	230	230	134	134
Missing	0	0	0	0	0	0
Mean	72.126	0.726	47.357	0.341	88.052	0.778
Std deviation	31.190	1.335	25.786	0.779	15.036	0.946
Skewness	−1.602	3.309	−0.259	6.767	−4.981	2.455
Std. error of skewness	0.240	0.240	0.160	0.160	0.209	0.209
Kurtosis	1.089	13.535	−0.965	61.041	26.133	6.769
Std. error of kurtosis	0.476	0.476	0.320	0.320	0.416	0.416
Range	99.342	8.103	99.558	8.613	97.305	5.289
Minimum	0.000	0.000	0.000	0.000	0.000	0.000
Maximum	99.342	8.103	99.558	8.613	97.305	5.289

The compression ratio distribution plot evidences a negative skewness (Skewness < 0) with leptokurtic kurtosis (Kurtosis > 0) except for the Beijing data where the kurtosis is slightly platykurtic (Kurtosis < 0). The curves for the three data sets are seen in Fig. 6.

The Skewness Error Distribution (SED) plot evidences a positive skewness (Skewness > 0) with leptokurtic kurtosis (Kurtosis > 0). The curves for the three data sets are shown in Fig. 7.

Pearson's correlation test between the compression ratio and the margin error (SED) for the Aracaju data set, shows a significantly positive correlation with $r = 0.322$ and p-value $= 0.001$, as shown in Table 6.

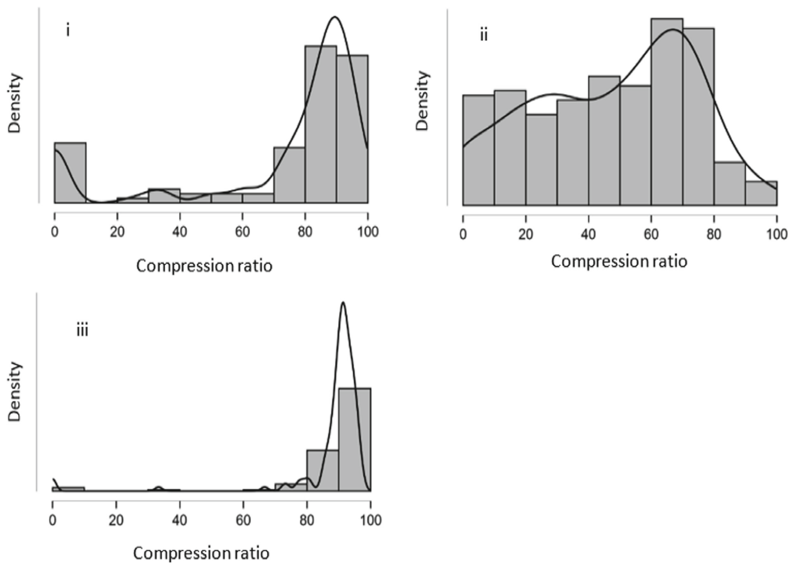

Fig. 6. Compression Ratio Distribution in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Algorithm TD-TR

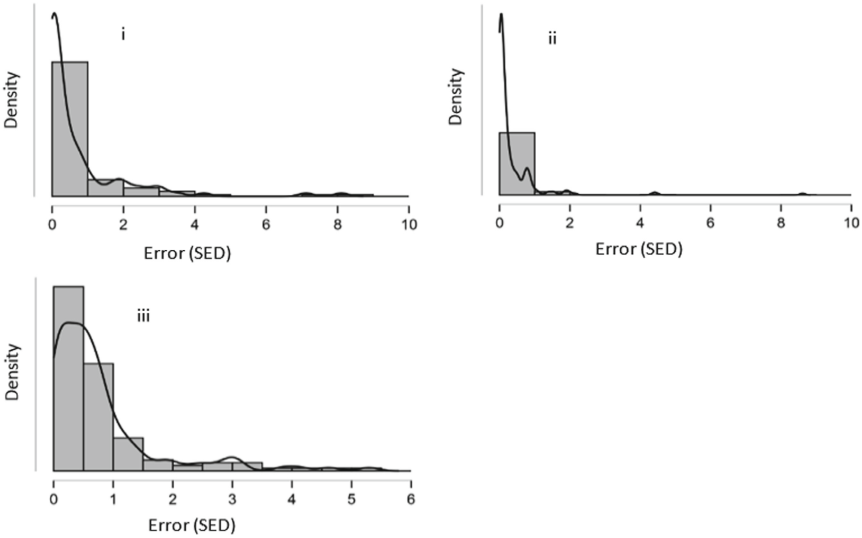

Fig. 7. Error Distribution (SED) (margin error) in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Algorithm TD-TR

Table 6. Pearson's Correlation for Compression Ratio and Error (SED) in the Aracaju data set - TD-TR Algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.322	n/a
	p-value	0.001	n/a

Pearson's correlation test between compression ratio and error (SED) for the Beijing data set evidenced a significantly positive correlation with $r = 0.424$ and p-value < 0.001, as shown in Table 7.

Table 7. Pearson's Correlation for Compression Ratio and Error (SED) on the Beijing data set - TD-TR Algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.424	n/a
	p-value	<.001	n/a

Pearson's correlation test between the compression ratio and the margin error (SED) for the Guayaquil data set, shows a significantly positive correlation with $r = 0.241$ and p-value $= 0.005$, as shown in Table 8.

Table 8. Pearson's Correlation for the Compression Ratio and Error (SED) in the Guayaquil data set - TD-TR algorithm

Variable		Compression ratio	SED
Compression ratio	Pearson's r	n/a	
	p-value	n/a	
SED	Pearson's r	0.241	n/a
	p-value	0.005	n/a

A visual inspection of the scatter plot, showed in Fig. 8, for the three data sets shows the correlation between compression ratio and margin of error.

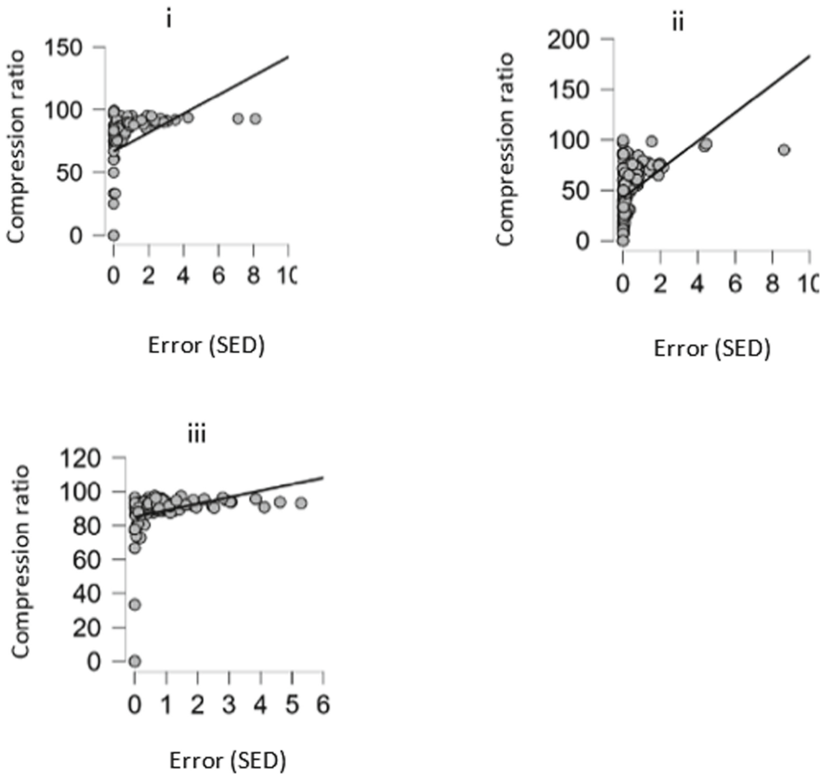

Fig. 8. Dispersion of the Correlation between Compression Ratio and Error (SED) (margin error) in the data sets (i) Aracaju (ii) Beijing and (iii) Guayaquil - Algorithm TD-TR

5 Analysis and Discussion

The results of this work reveal a significant positive correlation between compression ratio and error (SED) suggesting that the higher the compression ratio, the higher the error (SED). Three different data sets (obtaining different sample sizes randomly from different populations) were used on the two algorithms identified as relevant in the literature in order to provide sufficient and robust conclusions. After the experiments for the r indicator in Pearson's correlation, a mean of 0.312 for the Douglas Peucker algorithm and 0.329 for the TD-TR algorithm and a significant p-value level between 0.001 and 0.005 were obtained for both algorithms.

It is argued that the existing relationship between compression ratio and error (SED) when a point simplification algorithm processes a GPS trajectory is based on the fact that the results depend on the characteristic of the trajectory to be processed, so studies as in [27] use different simplification techniques on different datasets to make their studies consistent. On the other hand, removing points from an original trajectory implies an affectation in the quality of this and this is why multiple researches study how to improve the error levels when the compression ratio is improved [28].

The compression ratio by eliminating points from the trajectory to be simplified is decreasing the quality of the trajectory, therefore affecting its margin of error; the impact it causes on the quality will depend on whether the eliminated points are relevant or not. For certain data sets the simplification algorithms will remove relevant points which affects the margin of error. The main reason for the correlation between the two variables, in the authors' opinion, depends on the analysis of the data. In the case of the Douglas-Peucker algorithm, it performs a spatial analysis, while the TD-TR algorithm performs a spatio-temporal analysis, which allows simplifying points according to both their spatial and temporal location.

The main purpose of this work is to analyze the correlation between the compression ratio and margin of error metrics in the selected path simplification algorithms. As a result, it can be evidenced that the compression levels are given according to the characteristic of the data to be processed, so the compression levels vary from one dataset to another.

6 Conclusions and Future Work

The purpose of this work was to identify the relation between the compression ratio and the margin of error, which are indicators used by GPS trajectory simplification algorithms. The findings show a positive correlation between these two variables when trajectories are simplified, the value of the correlation should be interpreted depending on the level of compression of the algorithm when eliminating points considered not significant in a given context; likewise, the incidence of the margin of error and its effect when changing the compression ratio will depend on how significant the eliminated points are. GPS trajectory simplification algorithms seek to reduce the size of the trajectory to optimize storage space. In this sense, future studies can address mechanisms for these simplification algorithms to improve their compression ratio without affecting the error, i.e. that the correlation between these two variables is maintained or lower, demonstrating that the quality of the original trajectory would not be affected. Another important aspect would be to introduce a new variable to the correlational analysis such as the processing time of the algorithms.

References

1. Corcoran, P., Mooney, P., Huang, G.: Unsupervised trajectory compression. In: Proceedings - IEEE International Conference on Robotics and Automation, vol. 2016, pp. 3126–3132, June 2016. https://doi.org/10.1109/ICRA.2016.7487479
2. Ji, Y., Liu, H. , Liu, X., Ding, Y., Luo, W.: A comparison of road-network-constrained trajectory compression methods. In: Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS, pp. 256–263 (2017). https://doi.org/10.1109/ICPADS.2016.0042
3. Zheng, Y.: Trajectory data mining: an overview. ACM Trans. Intell. Syst. Technol. **6**(3), 1–41 (2015)
4. Muckell, J., Olsen, P.W., Hwang, J.H., Ravi, S.S., Lawson, C.T.: A framework for efficient and convenient evaluation of trajectory compression algorithms. In: Proceedings - 2013 4th International Conference on Computing for Geospatial Research and Application, COM.Geo 2013, pp. 24–31 (2013). https://doi.org/10.1109/COMGEO.2013.5

5. Salomon, D.: Data Compression: The Complete Reference. Springer, London (2007). https://doi.org/10.1007/978-1-84628-603-2
6. Gudmundsson, T.W.J., Katajainen, J., Merrick, D., Ong, C.: Compressing spatio-temporal trajectories. Comput. Geom. Theory Appl. **42**(9), 825–841 (2009)
7. Lv, C., Chen, F., Xu, Y., Song, J., Lv, P.: A trajectory compression algorithm based on non-uniform quantization. In: 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2015, pp. 2469–2474 (2016). https://doi.org/10.1109/FSKD.2015.7382342
8. Zheng, Y., Zhou, X.: Computing with Spatial Trajectories. Springer (2011). https://doi.org/10.1007/978-1-4614-1629-6
9. Lin, X., Ma, S., Zhang, H., Wo, T., Huai, J.: One-pass error bounded trajectory simplification. Proc. VLDB Endow. **10**(7), 841–852 (2017). https://doi.org/10.14778/3067421.3067432
10. Feldman, D., Sugaya, A., Rus, D.: An effective coreset compression algorithm for large scale sensor networks. In: IPSN'12 - Proceedings of the 11th International Conference on Information Processing in Sensor Networks, pp. 257–268 (2012). https://doi.org/10.1145/2185677.2185739
11. Hendawi, A.M., Khot, A., Rustum, A., Basalamah, A., Teredesai, A., Ali, M.: A map-matching aware framework for road network compression. In: Proceedings - IEEE International Conference on Mobile Data Management, vol. 1, pp. 307–310 (2015). https://doi.org/10.1109/MDM.2015.78
12. Song, R., Sun, W., Zheng, B., Zheng, Y.: A novel framework of trajectory compression in road networks. In: 40th International Conference on Very Large Data Bases, pp. 661–672 (2014). https://doi.org/10.14778/2732939.2732940
13. Van Hunnik, R.: Extensive comparison of trajectory simplification algorithms (2017)
14. Asif, M.T., Kannan, S., Dauwels, J., Jaillet, P.: Data compression techniques for urban traffic data. In: Proceedings of the 2013 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems, CIVTS 2013 - 2013 IEEE Symposium Series on Computational Intelligence, SSCI 2013, pp. 44–49 (2013). https://doi.org/10.1109/CIVTS.2013.6612288
15. Meratnia, N., de By, R.A.: Spatiotemporal compression techniques for moving point objects. In: Bertino, E., et al. (eds.) EDBT 2004. LNCS, vol. 2992, pp. 765–782. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24741-8_44
16. Lawson, C.T.: Compression and Mining of GPS Trace Data: New Techniques and Applications, New York (2011). http://www.utrc2.org/research/assets/202/Compression_and_Mining_of_GPS1.pdf
17. Reyes, G.: Algoritmo de compresión de trayectorias GPS basado en el algoritmo Top Down Time Ratio (TD-TR). In: 5to Congreso Científico Internacional Tecnología universidad sociedad, pp. 194–204 (2017)
18. de Vries, G.K.: Trajectory Compresion. University of Amsterdam (2011)
19. Sim, M., Kwak, J.-H., Lee, C.-H.: Fast shape matching algorithm based on the improved Douglas-Peucker algorithm. KIPS Trans. Softw. Data Eng. **5**(10), 497–502 (2016). https://doi.org/10.3745/ktsde.2016.5.10.497
20. Wu, S., Silva, A.C.G., Márquez, M.R.G.: The Douglas-Peucker algorithm : sufficiency conditions for non-self-intersections. J. Brazilian Comput. Soc. **9**, 1–17 (2004)
21. Wang, H.: Shark DB: An In-Memory Storage System for Large Scale Trajectory Data Management. The University of Queensland (2016)
22. Visvalingam, M., Whyatt, J.D.: Line generalisation by repeated elimination of the smallest area. In: Cartographic Information Systems Research Group, July 1992
23. Koegel, M., Baselt, D., Mauve, M., Scheuermann, B.: A comparison of vehicular trajectory encoding techniques. In: 2011 the 10th IFIP Annual Mediterranean Ad Hoc Networking Workshop, Med-Hoc-Net 2011, pp. 87–94 (2011). https://doi.org/10.1109/Med-Hoc-Net.2011.5970498

24. Zhang, S.K., Liu, Z.J., Cai, Y., Wu, Z.L., Shi, G.Y.: AIS trajectories simplification and threshold determination. J. Navig. **69**(4), 729–744 (2016). https://doi.org/10.1017/S0373463315000831
25. Hershberger, J., Snoeyink, J.: Speeding up the Douglas-Peucker line-simplification algorithm. ACM (1992)
26. Chen, M., Xu, M., Franti, P.: Compression of GPS trajectories. In: Data Compression Conference Proceedings, no. 61072146, pp. 62–71 (2012). https://doi.org/10.1109/DCC.2012.14
27. Zhang, D., Ding, M., Yang, D., Liu, Y., Fan, J., Shen, H.T.: Trajectory simplification: an experimental study and quality analysis. Proc. VLDB Endow., 934–946 (2018)
28. Muckell, J., Patil, V., Ping, F., Hwang, J.-H., Lawson, C.T., Ravi, S.S.: SQUISH: an online approach for GPS trajectory compression. In: Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications, 1–8 (2011)