

ORIGINAL PAPER

Open Access



# Application of Self-Organizing Maps on Time Series Data for identifying interpretable Driving Manoeuvres

Sivakkumaran Lakshminarayanan<sup>1,2</sup>

## Abstract

**Driving manoeuvre identification with Self-organizing Maps:** Understanding the usage of a product is essential for any manufacturer in particular for further development. Driving style of the driver is a significant factor in the usage of a city bus. This work proposes a new method to observe various driving manoeuvres in regular operations and identify the patterns in these manoeuvres. The significant advance in this method over other engineering approaches is the use of uncompressed data instead of transformations into certain Performance indicators. Here, the time series inputs were preserved and prepared as 10-second-frames using a sliding window technique and fed into Kohonen's Self-organizing Map (SOM) algorithm. This produced a high accuracy in the identification and classification of manoeuvres and at the same time to a highly interpretable solution that can be readily used for suggesting improvements.

**Driving Style Comparison:** The proposed method is applied to comparing the driving styles of two drivers driving in a similar environment; the differences are illustrated using frequency distributions of identified manoeuvres and then interpreted for the amelioration of fuel consumption.

**Keywords:** Self-organizing Map, Time Series, Driving Manoeuvres, Driving Style

## 1 Introduction

Driving Manoeuvres provide essential insights for automotive manufacturers to understand their vehicle usage and to improve their design. It is also useful for individual drivers as well as fleet owners to understand their vehicle usage to improve their operation and service.

**Approach** This work introduces a new method to identify and represent driving manoeuvres through data in conjunction with state of the art machine learning. There are several other data-driven approaches to predict or classify driving manoeuvres as stated in [15]. However, these methods are supervised and required labelled data to predict certain specific manoeuvres, such as exiting a round-about or stopping at a traffic light. The proposed

method is an unsupervised approach which requires no prior knowledge about the manoeuvres performed. This approach enables the learning of a wider spectrum of manoeuvres that are not captured in supervised approaches. The usage of Self-organizing Maps algorithm improves the interpretability of the results.

**Scope** This work uses time series data, precisely Velocity, and Accelerator and Brake pedal positions, to identify and classify driving manoeuvres. This type of data could be easily collected from any modern vehicle using a Controller Area Network (CAN) (as described in [5]) without any additional sensors or special equipment. The Fleet Management System (FMS) Standard [7] allows accessing this data, using an off-the-shelf connector, for trucks and buses from any European manufacturer. Accessing the CAN in a vehicle is usually not recommended since this might interfere with the operation of sensitive systems and also causes the warranty of the vehicle to cease.

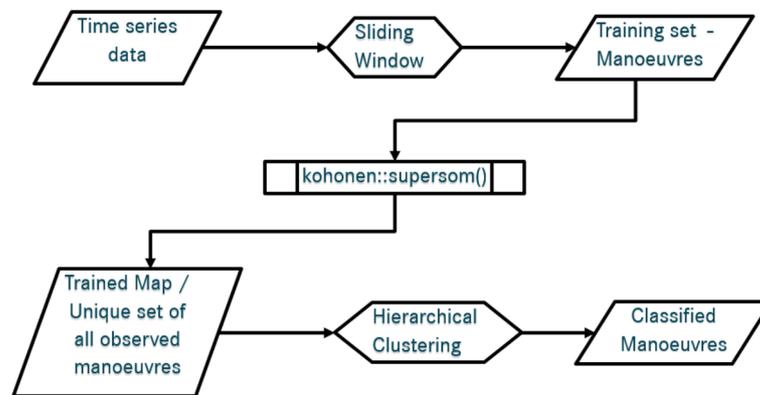
Correspondence: [sivakkumarann@gmail.com](mailto:sivakkumarann@gmail.com)

<sup>1</sup>Daimler Buses – EvoBus GmbH, 68305 Mannheim, Germany

<sup>2</sup>Goethe Universität Frankfurt am Main, Faculty of Computer Science and Mathematics, 60054 Frankfurt am Main, Germany



© The Author(s). 2020 **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.



**Fig. 1** Training Phase. Modelling and Classification of Manoeuvres

However, FMS CAN is explicitly defined for customers to access data, and it does not interfere with the vehicle functions nor affects the warranty of the vehicle. Thus, this input data selection improves the practicability of the proposed method.

The data considered in this work were collected for two years from city buses owned by a public transportation company in a particular city in Germany. The advantage of working on city buses is that they are operated systematically for long durations and distances. They also travel on almost all types of roads such as urban, suburban and even highways at times. They are operated at all times of the day, which includes dense traffic during the regular working hours of the city, to no traffic during midnights. Hence by nature, the collected data contains a wide variety of possible driving manoeuvres for the vehicle.

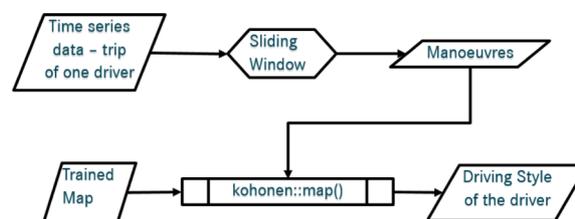
Since this work focuses specifically on city buses, many of the driving manoeuvres identified are quite specific to the same. For example, entering and leaving bus stops which are manoeuvres frequently performed only in city buses. Hence, the results can only be reproduced with city buses operated similarly. The proposed method can also be used for other vehicle types, but this is expected to produce a different set of driving manoeuvres.

**Application** In the latter part, using the driving manoeuvres identified, the driving styles of different drivers are

compared using the manoeuvres identified. Driving Style of a driver consists of various Driving manoeuvres performed with a given vehicle. Formally, this is the frequency distribution of all possible manoeuvres. Here, two drivers having different fuel consumption while driving the same vehicle on the same track are compared. A pair of similar manoeuvres performed by these drivers is presented as an example to explain the difference. This application also showcases the ease of use of the proposed method.

## 2 Methods

The workflow of the proposed method can be divided into 2 phases, namely Training phase and Characterization phase. In the Training phase, the time series data of several random trips driven by different drivers is taken as input, and then processed to identify a unique set of driving manoeuvres. Additionally, the manoeuvres are clustered for better interpretation. This process is described in detail in the following subsections, and Fig. 1 provides an overview. In the Characterization phase, time-series data of trips of an individual driver is taken as input, and then mapped onto the model trained in the previous phase, to classify the manoeuvres performed. The frequency distribution of the manoeuvres performed define the driving style of the driver. An overview of this process is shown in Fig. 2. In this work, individual trips of two different drivers



**Fig. 2** Characterization Phase. Mapping trips of individual drivers onto a pre-trained map

on the same track are mapped to compare their driving styles.

## 2.1 Data Preparation

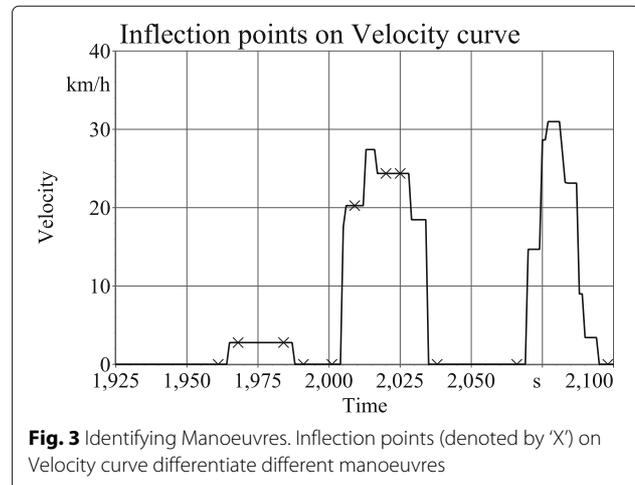
The data were collected using standard industrial data logging devices installed in vehicles with the consent of the respective customers. The data were collected at several time-frequencies, but for this work, 1Hz was sufficient. The variables considered are mentioned in Table 1.

**Sliding Window Preparation** The features (manoeuvres) had to be extracted from the time series data for modelling. The extracted time-frames are required to represent complete manoeuvres and have a fixed length for modelling. The inflection points (as described in [14]) on the velocity curve of the vehicle was considered as a reference point to differentiate manoeuvres as it distinguishes constant velocity and varying velocity segments of the curve. Due to the varying time durations between consecutive inflection points, as seen in Fig. 3, the segments themselves could not be utilized for modelling. Hence, the median of the time durations between consecutive inflection points was used as the standard duration of the time-frame. This value was determined to be 10 seconds (Rounded off).

When cutting the velocity curve into 10-second frames, it is possible to lose information. For instance, when a manoeuvre is shorter than 10 seconds, the beginning of the consecutive manoeuvre would become a part of the current frame. To avoid such information loss, the time series were processed by a sliding window, moving at a rate of 1 second. As the training phase requires considering all possible manoeuvres, the minimum possible moving rate seemed appropriate. In the characterization phase, higher moving rates will be used. All input variables were processed into 10-second frames, as explained above. However, distance and fuel consumption were aggregated for each frame. The data considered for modelling spanned 21 hours of operation, collected from the same vehicle through several random trips during weekdays in summer. The size of the data was limited to keep

**Table 1** Variables Used

Variables	Unit	Data preprocessing	Type
Velocity	m/s	sliding window	Vector of length 10
Accelerator Pedal Position	%	sliding window	Vector of length 10
Brake Pedal Position	%	sliding window	Vector of length 10
Fuel Consumed	ml/s	Sum on sliding window	Scalar
Distance Travelled	m	Sum on sliding window	Scalar



the computation manageable with a regular personal computer. The timespan was processed into sliding windows, and a randomly sampled 80% (60970 observations) were considered as the training set. The remaining 20% were stored as the test set for validation.

## 2.2 Training Phase

### 2.2.1 Modelling

**Kohonen's Self-organizing Maps** Kohonen's Self-organizing Map (SOM) algorithm was used to model this dataset and identify all the distinct manoeuvres observed. SOM is an Artificial Neural Networks algorithm that works based on competitive learning. Initially, a rectangular or hexagonal grid containing a fixed number of neurons or nodes is defined. The nodes are initialized with random vectors, having the same number of dimensions as the input data. The vectors corresponding to the nodes are collectively known as the codebook. During the training phase, input data points are randomly presented to the nodes, individually. All nodes compute their distance (usually Euclidean distance) to the given input and compete. The closest node is the winner or Best Match Unit (BMU). The neighbourhood surrounding the BMU with a certain radius, are considered as secondary winners. The BMU and its neighbourhood change their codebook vectors, adapting to the input, based on a learning function. The BMU tends to learn the most, while the neighbourhood learns comparatively less. In this way, the nodes spread throughout the data space. This is repeated for several iterations through the entire training set. Over the iterations, the learning rate and the neighbourhood radius decay, causing the nodes to stabilize. The final codebook thus contains centroids throughout the data space similar to K-means clustering.

After the training, the nodes can be visualized back in the original grid structure, known as a Map. This enables visualization of multidimensional data space as

low dimensional topology preserving maps. A detailed description of the algorithm can be found in [4, 6].

**Super-organized Maps** In this work, the variant of SOM known as Super-organized Map (supersom) was used. This variant allows the input variables to be grouped as layers, and the user can specify different weights for each layer. Here, the distance value is computed separately for each layer, and the learning is biased based on the weights. The implementation details of supersom can be found in [13]. The “kohonen” package [12] for R language was used to implement this algorithm.

**Training the map** In order to use supersom, the variables velocity, accelerator pedal position and brake pedal position (frame of 10 values) were each considered as separate layers. The fuel consumption and distance were grouped into a fourth layer. The four layers were given an equal weight of 0.25. This provides the importance of 12.5% to each scalar variable and 2.5% to the individual values in the vectors layers. The map was initialized with 400 nodes (20x20 grid) with a hexagonal grid structure. The number of training iterations was set as 700 because the mean distance between the inputs and their closest nodes did not change considerably on adding further iterations. The Training progress was observed to stabilize beyond 650 iterations and the mean distance to closest nodes was 72.7.

### 2.2.2 Clustering

Individually interpreting all of the nodes present on the map is possible, however painful. Clustering the nodes and interpreting the clusters would be a more manageable approach. To determine the number of meaningful clusters available on the map, Within-cluster sum of square distance ( $within_{SS}$ ) measure was tested iteratively between 2 to 20 clusters. The ‘elbow’ on the  $within_{SS}$  curve usually indicates the optimum number of clusters. In this case, it happens at 5, as shown in Fig. 4a. However, the curve does

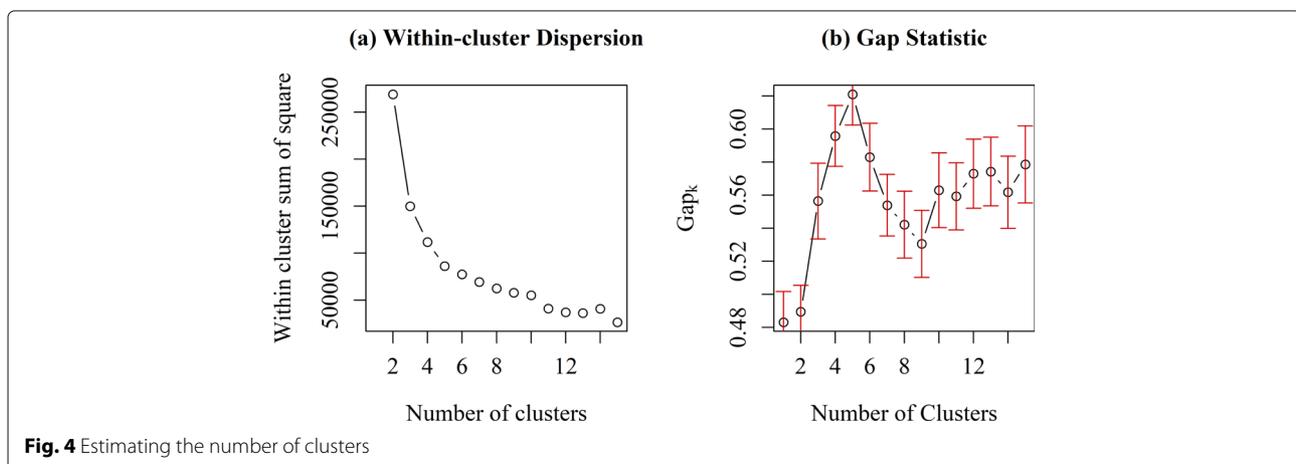
not monotonically decrease and flatten after 5. Hence, to make sure that 5 clusters are optimal, the Gap Statistic, as described in [11], was used additionally. Gap Statistic is the difference between  $\log(within_{SS})$  and its expectation under a null reference distribution of the data. In Fig. 4b, it can be observed that the Gap curve reaches the global maximum at 5. Therefore, the map was clustered into five groups for further interpretation.

### 2.3 Characterization Phase

In order to produce the driving style of a driver, complete trips are considered as input. A trip for a city bus is defined as operating the vehicle from an initial stop to an end stop and returning to the same initial stop. In terms of data collection, this is the period between an Engine-ON and the following Engine-OFF trigger. Within a trip, drivers do not change, as explained by the City Bus operators.

#### 2.3.1 Mapping trips to the pre-trained map

The mapping function works quite similar to the training function. Here, the input point is introduced to the pre-trained map, and the competition takes place. The BMU is straight away considered as the mapped node for this input. The codebook vectors do not change in this case. The test sets were mapped to the model with the help of the default “map” function provided by the same “kohonen” package. All manoeuvres of a driver’s trip are mapped onto the pre-trained model to obtain his/her Driving Style in the form of frequency distribution over all nodes on the SOM. The trips were preprocessed in the same way as the training set, using the sliding window approach explained earlier. However, the moving rate of the window was set as 10 instead of 1. Overlapping input frames are not required for characterization, because the need for completeness, i.e. of really capturing all possible driving actions in a trip, is not necessary. Only the training set needs to contain the overlapping frames in order not to miss any relevant physical manoeuvres, which would decrease the robustness



and applicability range of the model. The model should be able to handle any possible manoeuvre performed with the vehicle and regular operation.

### 2.3.2 Trip Comparison

Node densities for a given trip obtained from mapping are used for comparing the driving style across trips. In this work, two different trips driven with the same vehicle were considered. Since the driver identities are anonymous, the trips before and after the scheduled driver change were selected to make sure they were different drivers. The trips were individually mapped onto the pre-trained map. For simple interpretation, the difference between the trips was denoted as nodes common to both trips and nodes exclusive to either. Common nodes are manoeuvres that were observed on both trips, and they are considered as unavoidable manoeuvres due to the nature of the operational conditions. For example, waiting at the bus stop is unavoidable. Exclusive nodes are manoeuvres that are observed only in one of the trips. They depict the driving behaviour specific to a driver if observed multiple times in one trip and not observed in the other trip. In the current work, only the exclusive nodes are used to differentiate different trips. The normalized densities of nodes can also be considered to include the common nodes for distinguishing the trips. However, this is not performed within the current scope.

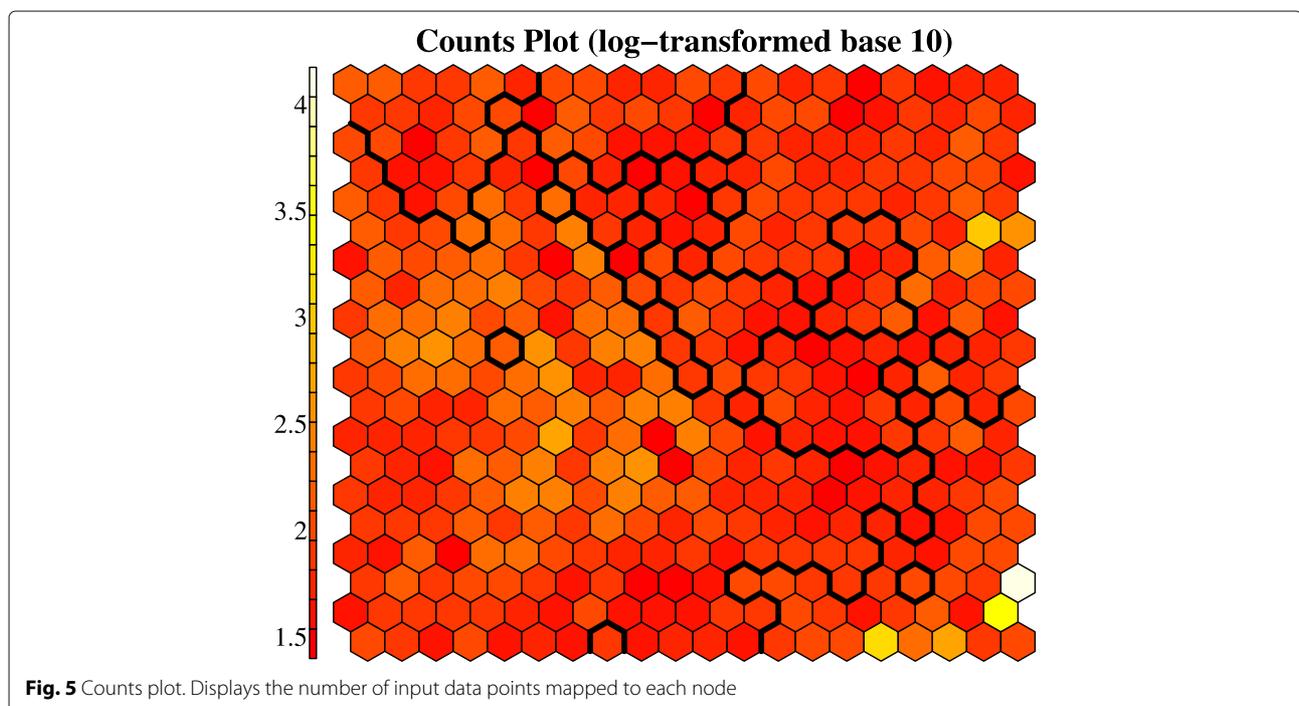
### 2.4 Interpreting Map Visualizations

The “kohonen” package offers functions to visualize any trained SOM model as Property Heatmaps or Code-maps.

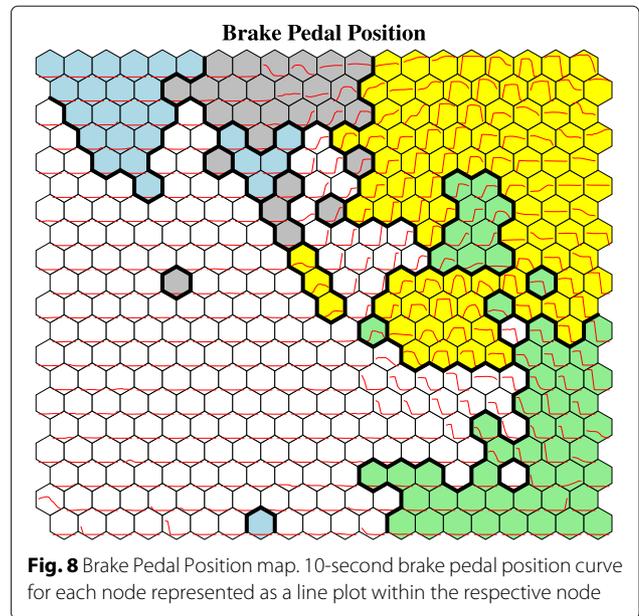
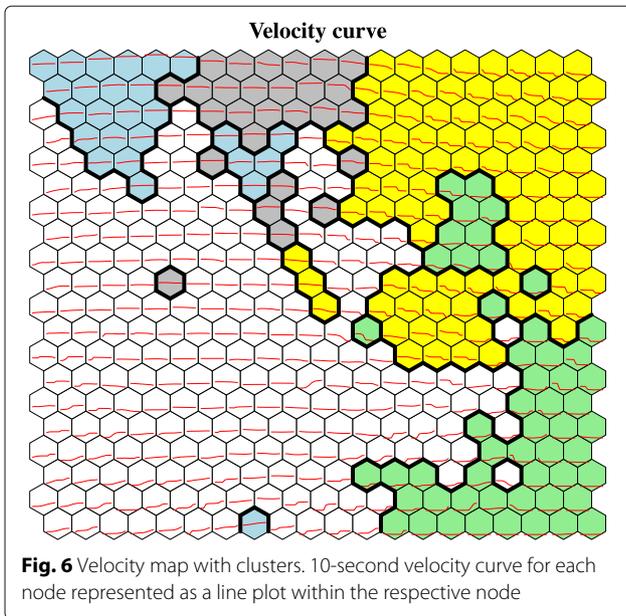
Regardless of the content or type of visualization, the ordering of nodes remains the same.

There are no X or Y axes for these maps. The nodes are numbered from 1 to 400. The 1st node is the bottom-left corner node, and the numbers increase from left to right. After the right edge, the numbering proceeds to the next row on top. Here, the bottom right node is, therefore, the 20th node and the 21st node is the upper left neighbour of the 1st node.

- Heatmaps visualize properties of the map, one at a time. The colour scale by default starts with Red for the minimum value and proceeds through Orange and Yellow, and finally ends at White for the maximum value. Grey colour represents Missing values or Null. The thick black lines represent the boundaries of the clusters. This scheme is used for Counts (Figs. 5, 10, 11), Fuel Consumption (Figs. 12, 13) and Trip Difference plots (Fig. 14).
- Code-maps visualize the codebook vectors of the map. Here, only one layer of the codebook can be visualized at once, regardless of the number of dimensions present in the layer. There are two representations used here.
  - For the time-frames, “Line” representation is used (Figs. 6, 7, 8). Here, the values are visualized as a 2D plot, with the corresponding variable and time as the Y and X-axes, respectively. The values are scaled to fit into the node, and all nodes have the same scale for



**Fig. 5** Counts plot. Displays the number of input data points mapped to each node



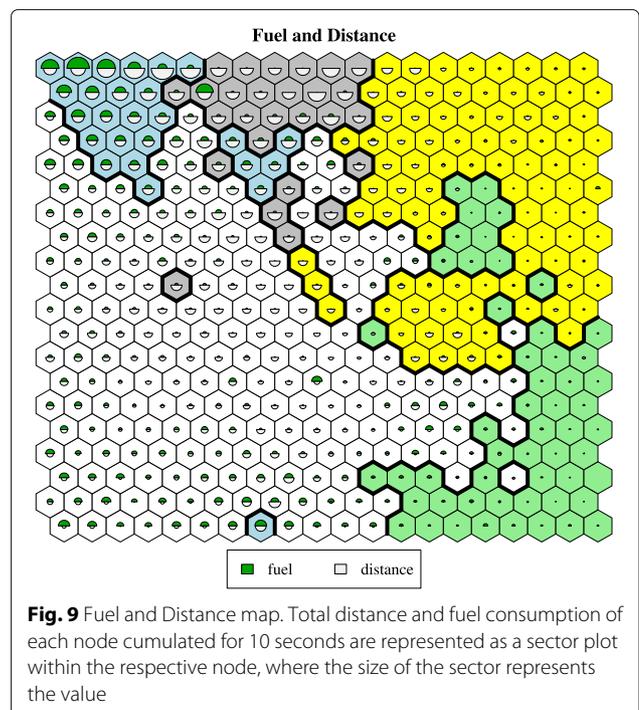
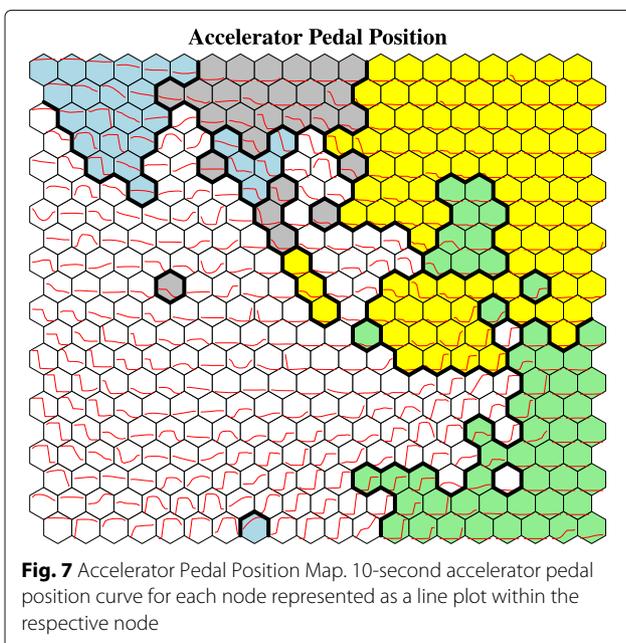
comparability. The axes are not marked, given the size of the nodes. But they can be still plotted separately from codebook vectors for better interpretation (Figs. 15, 16).

- For the variables in scalar layer, “Segments” representation is used (Fig. 9). Here a circle, which is split into equal sectors or segments, is present in each node. Each sector represents a variable, and the colouring convention is shown in the legend. The angle of the sectors is constant for all nodes (here, 180 degrees).

The size or radii of corresponding sectors vary with respect to the variable.

The background colour of the nodes in these maps represents their respective cluster.

**Counts Plot** This heatmap shows the number of input observations belonging to the respective node. The nodes had an average of 144.8 inputs mapped to each. Node 60



(3rd row from bottom, right edge) however had a mapping of 14947 observations, i.e., 24.5% of the training set. Upon verifying the codebook vector of the node, it was identified that the node represents the idle time of the vehicle with all values set as 0. This is normal for City buses since they spend more time at bus stops and traffic. Hence the observations were not excluded as outliers. For visualization, the mapping count was log-transformed with base 10 before plotting in Fig. 5.

The test set preserved earlier were also mapped onto the model for validation and were observed to have a similar distribution on the counts plot (Not shown here). The driving styles obtained in characterization phase are also presented as counts plot (Figs. 10, 11).

### 3 Results

#### 3.1 Interpreting the model and clusters

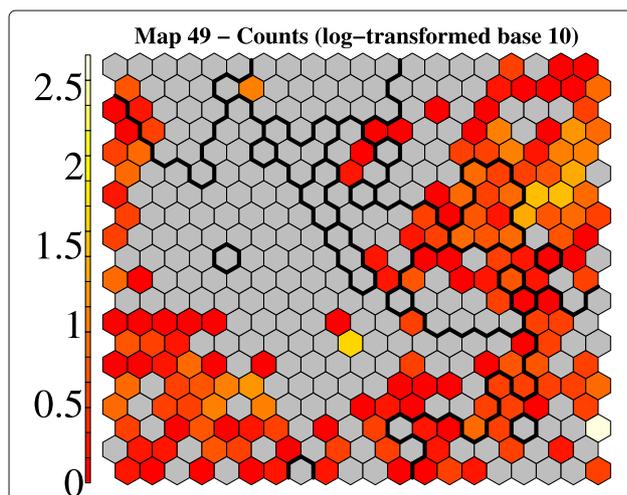
To understand the nodes on the map, the layers of the map and their clusters are interpreted below. In the end, the cluster interpretations across the layers are combined to provide the final interpretation for the cluster. These clusters are then used to provide the context for the manoeuvres.

##### 3.1.1 Velocity Map

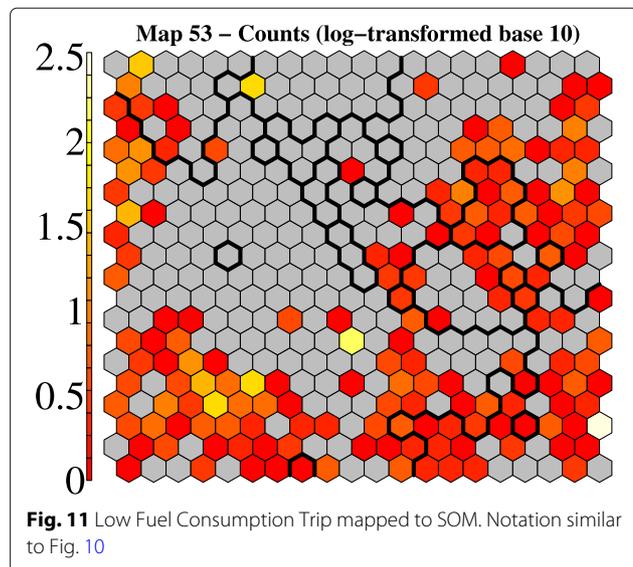
The code-map for Velocity layer is displayed in Fig. 6. The interpretation for each cluster is as follows:

**Blue** The velocities in these nodes increase gradually. Furthermore, they are always above the middle of the nodes, indicating above-average to high velocities.

**Grey** The velocities are decreasing gradually in these nodes. However, they appear to be close to the middle or higher, indicating high velocities.



**Fig. 10** High Fuel Consumption Trip mapped to SOM. Shows the mapping of the manoeuvres from the present trip to the map



**Fig. 11** Low Fuel Consumption Trip mapped to SOM. Notation similar to Fig. 10

**Yellow** The velocities are either decreasing rapidly towards zero or constantly zero.

**Green** The velocities are mostly at 0. Sometimes they decrease from a low velocity to zero or increase from 0 to a low velocity.

**White** These nodes contain a mixed set of velocity curves that are always higher than 0, however not as high as the blue or yellow cluster. The vehicle is not stopping in these nodes, but there are a few decelerations observed.

##### 3.1.2 Accelerator Pedal Position

The interpretation for accelerator pedal position layer is as follows (Fig. 7).

**Blue** The throttling behaviour appears to be aggressive in these nodes. The throttling is mostly high and sometimes released rapidly.

**Grey** The driver is mostly releasing the throttle and sometimes just 0%.

**Yellow** In most cases, there were no throttling observed. Few nodes have a rapid throttle from 0 or released to 0.

**Green** These nodes are very similar to the Grey cluster.

**White** The nodes cover all other behaviours observed. Constant 0% nodes are also present, however fewer than that of green or Yellow. Additionally, a few taps on the pedal were also observed.

##### 3.1.3 Brake Pedal Position

The observation of Brake pedal position variations (Fig. 8) is as follows.

**Blue** There was no braking in these nodes.

**Grey** There was no braking in half the nodes. In the rest, there was some gradual braking observed.

**Yellow** High brake usage was observed in these nodes, and most of them were rapid.

**Green** The brake pedal was completely 0% along the bottom. The remaining nodes have a brake release to 0 and the top most nodes have a rapid braking. However, the rapid press was still low than that of Grey cluster.

**White** Most of these nodes have no braking at all. A few nodes have rapid press or releases.

### 3.1.4 Distance covered and Fuel Consumed

Distance and fuel consumption are represented as sectors (Fig. 9). The observations are as follows.

**Blue** In comparison to other cluster nodes, these nodes have the highest fuel consumptions and distances covered.

**Grey** These nodes appear to travel distance slightly lesser than Blue cluster, but have very low or even 0 fuel consumption.

**Yellow** These nodes also have very low or zero fuel consumption. The distance covered is also low.

**Green** There is almost no distance covered in these nodes, but there are still small fuel consumptions observed.

**White** Low to average fuel consumption is observed. Distance covered is also low to average; however, they sometimes are not correlating at higher values of either.

### 3.1.5 Final interpretation of clusters

Based on the previous interpretations of the individual layer of the map, the final interpretations of the clusters are as follows.

**Blue** The driver had no intentions to stop the vehicle in the near vicinity, and the velocity is very high. Hence this cluster is termed as **High-Speed Zone**.

**Grey** The driver is slowing the vehicle, but not very rapidly. He is aware of a nearby stop or obstruction and hence is planning to stop gradually. Since the vehicle is running with less influence of throttle and brake, this cluster is termed as **Coasting Zone**. These nodes are good to have since they are very fuel-efficient.

**Yellow** These nodes also exhibit decelerations; however, they also have high braking. This implies the driver wants to stop the vehicle rapidly because of some circumstance and this manoeuvre is not fuel-efficient. The cluster shall be termed as **Rapid Deceleration Zone**.

**Green** This cluster shows gradual deceleration and acceleration close to 0. Node 60 and other nodes where the vehicle was standing most of the time were also present. Hence this cluster can be termed as **Bus Stop Zone**.

**White** The velocity in this cluster is average, and the braking and accelerations are random. The vehicle is also not stopping. Hence this cluster shall be termed as **Stop and Go Zone**.

## 3.2 Manoeuvres and Fuel Consumption

Two particular trips labelled Trip 49 and Trip 53 were considered for the characterization phase. Trips 49 and 53 had a fuel consumption of 16.1 litres and 7.8 litres, respectively, despite having travelled a similar distance of 40km approximately (40.13 km and 40.73km respectively), Trip 49 had more than twice the consumption of Trip 53. The mapping density is plotted in Figs. 10 and 11. The density is log-transformed similar to Fig. 5 due to the domination by node 60.

### 3.2.1 Trip 49 - High Consumption trip

Out of the 1335 manoeuvres 119 were not mapped to any of the map nodes. This might be because the training set consisted of a limited duration, and when a completely new driver is observed, the manoeuvres can be completely different.

As seen in Fig. 10, all types of manoeuvres (clusters) were observed in this trip. The node 60 had the highest density (number of mappings) with 501 observations mapped to it, similar to the training set. This being a bus stop zone manoeuvre, is an expected action in a city bus.

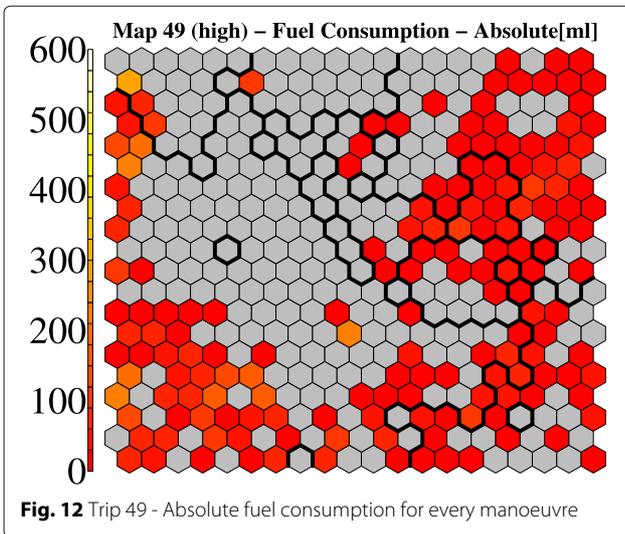
### 3.2.2 Trip 53 - Low Consumption trip

Since the trips were having the same time duration, trip 53 also had 1335 manoeuvres observed, out of which 120 were not mapped to any nodes.

It was observed that most of the mappings of trip 53 were also quite similar to trip 49, as seen in Fig. 11. Node 60 was also the highest density node in the trip. In comparison to trip 49, trip 53 had fewer types of Yellow or rapid deceleration manoeuvres present on the top-right of the map.

### 3.2.3 Trip differences

When observing the fuel consumption of the trips in Figs. 12 and 13, Trip 53 was found to have more fuel consumption in the High-Speed Zone cluster and Bus Stop

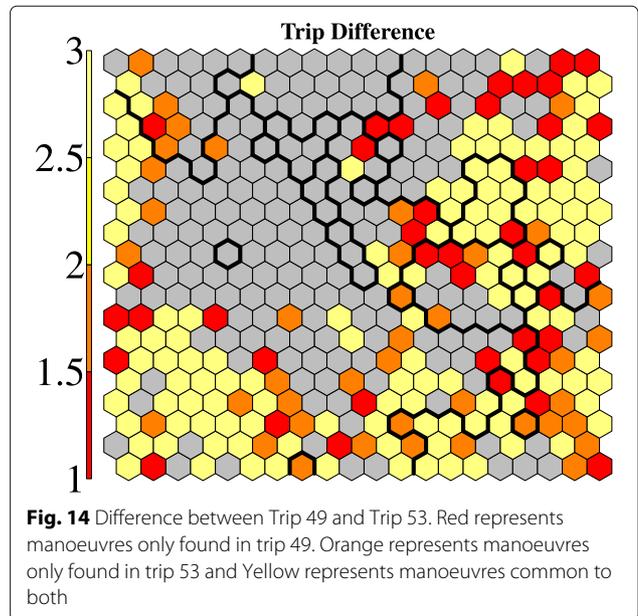
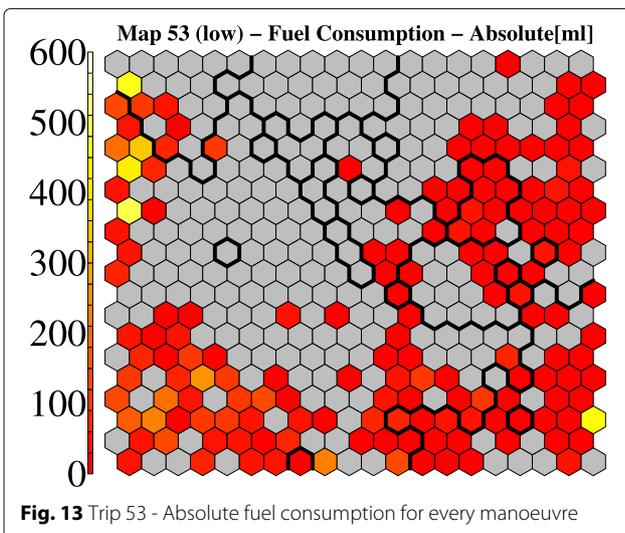


Zone node. Trip 49 did not have such very high consumption manoeuvres. On the contrary, Trip 53 had less fuel consumption for the total trip.

In order to distinguish the manoeuvres observed in the two trips Fig. 14 was constructed where, 1 indicates that the manoeuvre was exclusive to Trip 49, 2 indicates that the manoeuvre was exclusive to Trip 53, and 3 denotes it common to both trips.

In Fig. 14, nodes 78 and 79 (Yellow node on the right edge, fourth from the bottom of the map in Fig. 14 is node 80. Its neighbours to the left are 78 and 79), are coloured red and orange respectively. These are nodes present in the Bus Stop Zone or Green cluster. Red denotes, it was observed only in Trip 49 and orange was only present in Trip 53.

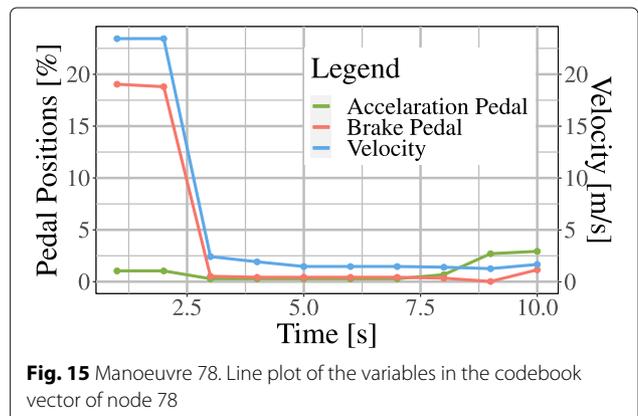
To investigate the driving manoeuvres better, a pair of similar manoeuvres is taken. The codes of node 78 and

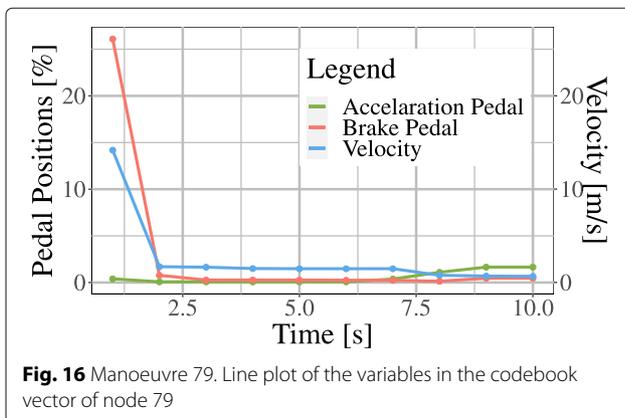


79 are plotted in Figs. 15 and 16, respectively. In node 79, the velocity was initially less than 15 m/s and due to the braking, reduces to approximately 2 m/s. The brake was released at time 2 s and the velocity was constant until time 7 s. To avoid halting, Accelerator Pedal was slightly pressed.

In node 78, the velocity was initially higher at about 24 m/s. Due to the braking, it decelerated to approximately 2 m/s at time 3 s and dropped further until time 5 s. Again the Accelerator Pedal was pressed here, slightly higher than in node 79.

Thus manoeuvre 78 was more aggressive than 79. It can be concluded that the driver of Trip 49 was more aggressive at lower velocity manoeuvres when compared to the driver of Trip 53. This explains the fuel consumption difference to some extent.





#### 4 Discussion

**Time Series with SOM** The significant contribution of this work is the approach of using SOM to represent driving manoeuvres. [1] describes the advantages of using SOM for Time Series Prediction problem, and emphasizes the Local Nature and Topology Preserving properties of SOM-based models. The current work extends the same idea over a multivariate scenario. The topology-preserving property enables the identification of comparable driving manoeuvres. Due to the data preparation, the codebook vector of the model provides the properties of the map as time series, which makes the results interpretable. Hence, the proposed method is more comfortable to implement and interpret compared to the state of art approaches summarized in [15].

**Driving Manoeuvres vs Fuel Consumption** When it comes to fuel consumption optimization concerning driving styles, a common approach followed is to build a speed profile, similar to [8] or use of metrics such as Vehicle Specific Power similar to [2, 3]. Although these approaches are quite robust and effective, it is quite difficult to communicate the insights to the drivers, and the context of the driving behaviour is lost. The proposed driving manoeuvre definition with velocity, throttle and brake pedal positions provides an easily interpretable method to reduce fuel consumption.

**Driving Style** It is also quite common to distinguish driving styles as Aggressive, Normal, Gentle, and so on, as performed in [10]. In the case of City Buses, there are more restrictions due to fixed schedules and traffic. Aggressiveness and defensiveness cannot be easily computed, and a driver can exhibit both depending on the context. For instance, a driver can be aggressive while entering a bus stop and defensive while leaving. [9] shows the impact of driving styles on fuel consumption at specific points of a trip, such as bus stops and roundabouts.

The proposed method does not distinguish the driving style as aggressive or defensive and rather differentiates driving manoeuvres. This would help drivers learn where they are inefficient and improve on those manoeuvres precisely. Furthermore, the driver could also be learning from his manoeuvres from a different time. Due to the regularity in the trips and localized nature of SOM, the clusters identified could easily specify the context of the manoeuvre such as bus stop even without a controlled measurement environment as done in [9].

#### 5 Conclusion

A method for processing time-series data with SOM to define driving manoeuvres has been introduced. The SOM nodes represent a possible set of driving manoeuvres based on the observed fleet. The clusters identified from the SOM nodes provide the context for the driving manoeuvres and help in understanding the usage of the vehicle. With the help of this model, the fuel consumption of two different trips on the same track was compared. The Bus-Stop manoeuvre of the trips was used as an example to compare similar manoeuvres. This can be built into an application for driver feedback for fuel optimization.

The method introduced used velocity and pedal positions to define the manoeuvres so that the results and interpretations are conveyable to common users like bus drivers. The method can be extended further by using different input and target data, and the resulting mappings of the trip on the SOM can also be used as inputs to further use-cases, which were not discussed in this article, but currently being developed.

##### Author's contribution

SL wrote the entire article and all the necessary steps in the study, starting from Data preparation until the interpretation of the results were performed by the same. The author(s) read and approved the final manuscript.

##### Funding

EvoBus GmbH has funded this work entirely. There has been no influence by EvoBus GmbH or any other organizations, in the analysis or the results presented here.

##### Availability of data and Material

The data that support the findings of this study were obtained from vehicles manufactured by EvoBus GmbH and operated by public transport companies in Germany. Restrictions apply to the availability of these data, and they are not publicly available. Data are however available from the author upon reasonable request and with permission of EvoBus GmbH.

##### Competing interests

This work was facilitated by EvoBus GmbH both in terms of Data and Funding. The author is employed full time as a PhD Candidate by the same organization.

Received: 16 July 2019 Revised: 14 April 2020 Accepted: 16 April 2020

Published online: 29 April 2020

##### References

1. Barreto, G.A. (2007). Time series prediction with the self-organizing map: A review, In *Perspectives of Neural-Symbolic Integration*. [https://doi.org/10.1007/978-3-540-73954-8\\_6](https://doi.org/10.1007/978-3-540-73954-8_6) (pp. 135–158). Berlin: Springer.

2. Carrese, S., Gemma, A., La Spada, S. (2013). Impacts of driving behaviours, slope and vehicle load factor on bus fuel consumption and emissions: a real case study in the city of rome. *Procedia-Social and Behavioral Sciences*, 87, 211–221.
3. Frey, H.C., Roupail, N.M., Zhai, H., Farias, T.L., Gonçalves, G.A. (2007). Comparing real-world fuel consumption for diesel-and hydrogen-fueled transit buses and implication for emissions. *Transportation Research Part D: Transport and Environment*, 12(4), 281–291.
4. Hastie, T., Tibshirani, R., Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction*. New York: Springer Science & Business Media. isbn=9780387848587, <http://citeseerx.ist.psu.edu/viewdoc/download?https://doi.org/10.1.1.158.8831&rep=rep1&type=pdf>. Accessed 24 Apr 2020.
5. HPL SC (2002). Introduction to the controller area network (can). *Application Report SLOA101*, 1–17. Texas Instruments, <http://www.ti.com/lit/zip/sloa101.pdf>.
6. Kohonen, T. (1990). The self organizing maps. *Proceedings of IEEE*, 78, 1464–1480. <https://doi.org/10.1109/5.58325>. <https://ieeexplore.ieee.org/document/58325>. Accessed 24 Apr 2020.
7. LogiCom GmbH (2017). Fms-standard . <http://www.fms-standard.com/Bus/faq.htm>. kohonenPaper.
8. Nouveliere, L., Braci, M., Menhour, L., Luu, H., Mammar, S. (2008). Fuel consumption optimization for a city bus, In *UKACC Control conference*. (pp. 1–6), isbn=978-0-9556152-1-4. <http://ukacc.group.shef.ac.uk/proceedings/control2008/papers/p50.pdf>. Accessed 24 Apr 2020.
9. Rohani, M., & Buhari, R. (2014). How much money can be saved? impact of driving style on bus fuel consumption, In *InCIEC 2013*. [https://doi.org/10.1007/978-981-4585-02-6\\_35](https://doi.org/10.1007/978-981-4585-02-6_35) (pp. 399–411): Springer.
10. Stanton, N. (2019). *Driving Style Modelling with Adaptive Neuro-Fuzzy Inference System and Real Driving Data*, (pp. 481–90). Cham: Springer.
11. Tibshirani, R., Walther, G., Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411–423.
12. Wehrens, R., & Buydens, L.M.C. (2007). Self- and super-organizing maps in R: The kohonen package. *Journal of Statistical Software*, 21(5), 1–19. <https://doi.org/10.18637/jss.v021.i05>.
13. Wehrens, R., & Kruisselbrink, J. (2018). Flexible self-organizing maps in kohonen 3.0. *Journal of Statistical Software, Articles*, 87(7), 1–18. <https://doi.org/10.18637/jss.v087.i07>. <https://www.jstatsoft.org/v087/i07>. Accessed 24 Apr 2020.
14. Weisstein, E.W. (2019). Inflection point. <http://mathworld.wolfram.com/InflectionPoint.html>. Accessed 24 Apr 2020.
15. Zhao, M. (2019). Modeling driving behavior at single-lane roundabouts. PhD thesis. <https://doi.org/10.24355/dbbs.084-201903071236-0>. [https://publikationsserver.tu-braunschweig.de/receive/dbbs\\_mods\\_00066445](https://publikationsserver.tu-braunschweig.de/receive/dbbs_mods_00066445). Accessed 24 Apr 2020.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)