2019 IEEE Intelligent Transportation Systems Conference (ITSC)
Auckland, NZ, October 27-30, 2019

Iterative DBSCAN (I-DBSCAN) to Identify Aggressive Driving Behaviors within Unlabeled Real-World Driving Data

Charles Marks, Arash Jahangiri, *Member, IEEE*, Sahar Ghanipoor Machiani

Abstract— **Each year, 1.35 million people die and over 50 million are injured in traffic accidents. Over half of fatal accidents are due to aggressive driving behaviors. Machine learning analytic strategies hold promise in helping to identify aggressive driving behaviors within real world driving (RWD) datasets, but innovative strategies are required in order to achieve this promise. Herein, we introduce and define Iterative DBSCAN (I-DBSCAN), an extension of the Density Based Spatial Clustering of Applications with Noise algorithm, as one tool that can be utilized as part of a machine learning analytic strategy for identifying aggressive driving behaviors within large, unlabeled RWD datasets. Further, we provide a case example of I-DBSCAN's application and discuss how its application can enhance efforts to identify aggressive driving and improve overall traffic safety.**

I. Introduction

Improving traffic safety is of growing importance. Each year, 1.35 million die and over 50 million are injured in traffic accidents [1]. Further, the AAA Foundation for Traffic Safety estimate that over half of all fatal traffic incidents are the result of aggressive driving behaviors [2]. Strategies which show promise in identifying incidents of aggressive driving have the potential to greatly improve overall traffic safety, reduce road crash injuries, and fatalities globally.

With the increased ability to collect real-world driving (RWD) data, machine learning analytic strategies can and must be utilized to improve overall traffic safety. While the availability of such RWD data is a boon for traffic safety endeavors, the challenge of identifying aggressive driving behaviors in large (often poorly or variably structured) sets of unlabeled RWD data is a complex challenge which requires innovative strategies. These approaches must be flexible enough to handle variations in data collection methods and structure, fast enough to be reasonably executed by traffic safety organizations, and provide interpretable and actionable results for improving overall traffic safety.

Charles Marks is with the School of Social Work, San Diego State University, San Diego, CA 92182 USA (e-mail: cmarks@sdsu.edu).

Arash Jahangiri is with the Department of Civil, Construction, and Environmental Engineering, San Diego State University, San Diego, CA, 92182 USA (corresponding author, phone: 619-594-1937; e-mail: ajahangiri@sdsu.edu).

Sahar Ghanipoor Machiani is with the Department of Civil, Construction, and Environmental Engineering, San Diego State University, San Diego, CA, 92182 USA (e-mail: sghanipoor@sdsu.edu).

Herein, we shall introduce the implementation of the algorithm Iterative DBSCAN (I-DBSCAN) as a classification method for identifying aggressive driving events from unlabeled RWD data. The goal of I-DBSCAN is to return a small subset of outlying driving observations from RWD data which we shall refer to as "abnormal" driving behaviors. This small subset of abnormal driving behaviors can then be understood as the potential pool of aggressive driving behaviors. We shall define the algorithm, provide a case example of how it can be implemented, and discuss the role I-DBSCAN can play in efforts to identify aggressive driving and improve traffic safety.

II. Algorithm

A. Overview

Herein, we propose Iterative DBSCAN (I-DBSCAN) as an algorithm for identifying abnormal (and, thus, potentially aggressive) driving behaviors within an unlabeled RWD dataset. The algorithm can be broken into three general steps: (1) formatting the data; (2) dividing the data into "elementary" subsets; and, (3) running I-DBSCAN on each "elementary" subset, extracting noise and small outlying clusters.

B. Formatting the Data

The first step is to format the real-world driving data from "time-point" (TP) data into "monitoring period" (MP) data.

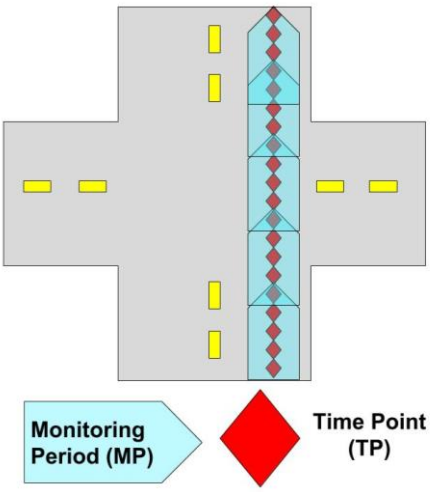

Figure 1. Converting TP Data to MP Data

The dataset we utilized contained the movement data for vehicles by returning a set of variables (e.g. speed, acceleration, yaw rate, etc) every deci-second. We then

generate a new dataset where each observation is an MP with a duration of *x* seconds occurring at intervals of *y* seconds. To do this, first sort and group TP data by trip (i.e. each grouping represents one continuous driving trip from start to finish) and then convert these groupings of TPs into corresponding MPs. For example, we can format data such that our MPs have a duration of 3 seconds and occur at 1 second intervals (see **Figure 1**). Since our data contains TPs at deci-second intervals, this means each MP is representative of 30 TP observations from our data. Each MP will then contain mean, minimum, maximum, and standard deviation values for each of the variables available in the TP dataset, as well as information about change in heading and, when available, other datum such as road type or speed limit. As well, it should be noted, in the example described, MPs will overlap one another.

C. Sub-setting the Data

Prior to applying I-DBSCAN, it is necessary to subset the MP data into "elementary" subsets. By "elementary", we mean that there exist certain foundational driving behaviors (such as making left turns, right turns, and merging, among others) and that we seek to subset our MP data such that all observations in a given subset are reflective of one elementary driving behavior (EDB). Two steps can be taken to divide the MP data into EDB subsets. The first step is to subset the data by physical environment. The spectrum of driving behaviors that may occur at a location are different depending on the road class (e.g. a local road versus a freeway), neighborhood, and traffic regulations (e.g. speed limits, traffic signs, stop lights). If road class and traffic regulation data are available, this information can be used to subset MP data. Otherwise, coordinate bounding boxes of desired environment can be utilized to subset the MP data. The second step is to then subset MP data by "behavioral factors" (i.e. turning, accelerating, coasting, etc.) and this can either be done by utilizing observation constraints (such as extracting left turns by sub-setting the MP data based on change in vehicle heading) or can be done by utilizing the k-means clustering method. While we shall focus on applying observation constraints to generate our data subset for the next step, k-means could be utilized by running the clustering algorithm on the MP data and sub-setting it by cluster identified. Each cluster of MP data could then be utilized in the next step.

D. Running I-DBSCAN and Identifying Abnormal Driving

I-DBSCAN utilizes the DBSCAN (Density Based Spatial Clustering of Applications with Noise) clustering algorithm, a method for identifying arbitrarily shaped clusters within spatial data defined by Ester et al [3]. DBSCAN requires two inputs, minPts and ε, such that ε is the radius of a neighborhood around a given point and a point *p* is said to be a *core point* if minPts other points are within ε of that point. Clusters are formed by grouping adjacent core points and *border points* (i.e. points within ε of a core point but which do not have enough ε-neighbors to be a core point). Points which do not meet the definition of a core point or border point are left unclustered and considered to be *noise*. Concerns about running DBSCAN with high-dimensional data have been raised ("curse of dimensionality") [4], thus, depending on the number of variables being used to cluster on, a principle component analysis (PCA) can be utilized to reduce data dimensionality.

At this stage we shall apply I-DBSCAN to the MP data subset to extract observations of abnormal driving behavior. One iteration of I-DBSCAN involves the following steps: (optional) run a principal component analysis on the data to reduce dimensionality, (1) identify ε and minPts parameters for running DBSCAN on the data, (2) run DBSCAN and ensure one cluster (the normal cluster) compromises at least *normPercent* of the data (this is the normal EDB cluster), (3) extract the normal cluster, any abnormal clusters, and noise, and (4) determine whether the procedure is complete or to repeat the steps with just the normal EDB cluster.

The first step is to identify the ε and minPts parameters for running DBSCAN. The minPts parameter can be set at twice the number of dimensions being clustered on [5]. Once the minPts parameter is chosen, the ε value can be chosen by identifying the "elbow" of a k-nearest neighbors distance plot utilizing the chosen minPts parameter [6].

Once the ε and minPts parameters are chosen, the second step is to run DBSCAN. Because in prior steps the data has been subsetted as to reflect a single EDB, the dataset should be very dense. While in many cases it is considered a "red flag" [6] when DBSCAN returns nearly all of the observations in one cluster, in this case, that is in fact the goal. It is, thus, important to establish a threshold percentage *normPercent* (while we have opted for a default value of 90%, further investigation can be done to determine an optimal value). If there is a cluster whose observations account for at least *normPerc*ent of the data clustered, then this cluster is deemed the normal EDB cluster. If there does not exist such a cluster, then multiple options are available: the first is to increase ε, which will work to decrease the number of clusters identified [6]; the second is to decrease *normPercent*; and the third is to return to prior steps to ensure that the data is representative of a singular EDB.

Then, the third step: The normal cluster is extracted into a new dataset -- if I-DBSCAN is run for another iteration, this is the dataset that shall be used. As well, noise and any other clusters identified shall be extracted and included in our final set of abnormal observations.

The final step is to determine whether or not to execute another iteration of I-DBSCAN. There are two ways to terminate running I-DBSCAN. The first is that if an iteration does not return a normal cluster of at least *normPercent* of the total data, even after adjusting parameters,[1] then the algorithm should be terminated and no results should be extracted. In this case, the prior step (sub-setting the data) should be redone to ensure the data is representative of an EDB. The second, ideal, way for I-DBSCAN to terminate is by reaching the *stopThresh* threshold. If *stopThresh* iterations occur which only return a normal cluster and noise (i.e. DBSCAN does not

[1] Note that if *NormPercent* is dropped too low (say below 75%), it defeats the purpose of I-DBSCAN. ε can be increased as the user sees fit and it will be important to address decisions for modifying this variable in the interpretation of results.

return any abnormal clusters), then I-DBSCAN is terminated and the results are returned. As noise and abnormal clusters are removed from the dataset, the ε value for each iteration will change. This change in ε will ensure that in each subsequent iteration we identify additional abnormal observations.

```
1: Algorithm I-DBSCAN(data, normPercent, stopThresh):

2:    abnormal = empty list ()
3:    stopCounter = 0
4:    minPts = dimensions(data)*2

5:    while(stopCounter < stopThresh), do
6:          ε = identifyEps(data, minPts)
7:          results = DBSCAN(data, minPts, ε)
8:          if( cluster exists that is normPercent of data )
9:             normal = that cluster
10:         else
11:            throw error (change parameters)
12:         abnormal = abnormal + (data – normal)
13:         if( only one cluster found )
14:            stopCounter++
17:         data = normal

18:   return abnormal
```

E. Rationale for I-DBSCAN

While defining how the results of I-DBSCAN may be interpreted is an important next task, it is important to discuss why, exactly, I-DBSCAN is potentially useful in identifying abnormal driving behaviors. First, the utilization of I-DBSCAN is based on the idea that abnormal driving behaviors (and of special interest, aggressive driving behaviors) will appear, based on the available metrics, similar to their normal counterparts – for example, an abnormal (or aggressive) right turn is still a right turn and must first be identified as a right turn before it can be identified as being abnormal in some way. This is why, before running I-DBSCAN, it is necessary to subdivide the MP data into EDB subsets. If we were to forgo this step and run I-DBSCAN on a dataset which contained a variety of EDBs, the algorithm would likely cluster the data based on EDBs, not on abnormality – for example, if the data contained right and left turns, the data pertaining to the differences in turn direction (such as yaw rate) would dominate the clustering and instead of identifying abnormal observations we would simply be identifying right turns versus left turns.

Further, the utilization of I-DBSCAN is based on the notion that abnormal variants of normal EDBs will not necessarily be similar to one another and that there is no guarantee that there is a cluster which they may be a part of – for example, if we are running I-DBSCAN on a right turn only subset, both an incident where the driver sped through the turn and an incident where the driver slammed on the brakes halfway through the turn may be identified as abnormal, but, these observations may be less likely to be clustered to one another than they are to be clustered within the normal EDB cluster. Simultaneously, if small clusters of similar abnormal driving exist, we want to be able to identify those, as well, because they indicate patterns of abnormal or aggressive driving.

As such, I-DBSCAN is an ideal algorithm for seeking to identify abnormal driving behaviors. First, because the data fed to I-DBSCAN represent a single EDB, the dense nature of the data will result in I-DBSCAN clustering a majority of the data within a single cluster (this is the normal EDB cluster). Second, if there exist small clusters of outliers, I-DBSCAN will be able to identify these as potential clusters of abnormal driving behavior. Finally, I-DBSCAN will also identify noise. These observations are those that were not similar enough to the normal EDB cluster to be included but, also, do not have a dense enough set of similar observations to be included in any cluster. By running DBSCAN iteratively in this manner, we are able to identify both clusters of abnormal driving behaviors in addition to singular such incidents. The iterations of I-DBSCAN are akin to peeling away the dry layers of an onion, you may pull the first layer back only to find a second layer in need of peeling. It is through this iterative process, that the "layers" of abnormal observations are "peeled away" from the normal cluster within.

III. Case Example

Here we shall implement I-DBSCAN on BSMP1 data. We shall look at the right turn EDB data for a single intersection (Huron and Main Streets) in Ann Arbor, MI, apply I-DBSCAN, and examine the results. Data was stored in a PostGreSQL database and all analyses were conducted using R (packages utilized: dbscan, shiny, FactoMineR, tidyverse, RPostgreSQL, postGIStools) and were run on an Amazon EC2 t2.micro instance.

A. Dataset

Data were collected as part of the Safety Pilot Model Deployment (SPMD) study at the University of Michigan which collected RWD data in Ann Arbor, MI [7]. SPMD was conducted by equipping approximately 3,000 vehicles in the Ann Arbor area with technology that, among other things, recorded TP observations of the vehicle's location, speed, acceleration, yaw rate, and other movement based statistics at a deci-second interval [7] [8]. Specifically, we utilized all observations from April, 2013 from the BSMP1 dataset (which is publicly available here).

The April BSMP1 dataset is over one-half of a terabyte in size. While in the algorithm definition we describe sub-setting the data by environment after formatting the data, due to the large size of the dataset, we opted to subset this dataset by longitude and latitude prior to formatting as to only include data corresponding to the intersection of Huron and Main. This was done by creating a new table in our PostGreSQL database. This new dataset contained 1,221,060 observations.

B. Formatting the Data

Our dataset was then loaded into RStudio. The first step was then to convert the TP data into MP data. For this, we opted

to use an MP of 3 seconds at 1 second intervals. It should be noted that because the TP data is recorded at deci-second intervals and since we are converting it into MP data at 1 second intervals, that this effectively shrank the size of our dataset by an order of magnitude. After formatting, the MP dataset contained 124,896 observations.

C. Sub-setting the Data

As the data had already been sub-setted by environment (intersection of Huron and Main), the next step was to divide the data into EDB subsets. For this analysis we wanted to examine right hand turns at this intersection. Each MP observation included the variable *Change in Heading*, which defines the change in vehicle direction from the start of the monitoring period to the end – this was done by taking the modular difference between the heading at the end of the MP and at the start of the MP. To extract right turns, we included all observations for which the recorded change in heading was equal to or greater than 45 degrees. After extraction, our data contained 1,231 observations representing a total of 495 distinct right turns (i.e. there are multiple observations corresponding to individual turn events).

A couple notes on considerations for implementation strategies. The first is that utilizing I-DBSCAN is dependent on being able to define an elementary driving behavior (EDB), which is, in itself, a complex task. We offer identifying an environment (such as an intersection) and movement type (such as a right turn) as an effective way of identifying EDBs, but the process may need to be altered in order to identify EDBs of interest. Further, there may be EDBs that are not easily defined by simple parameter restriction but that a clustering method such as k-means might pick up. All of this is to emphasize that the process of sub-setting the data is just as important as the implementation of I-DBSCAN itself.

D. Running I-DBSCAN

We started with 14 variables to cluster on (average, minimum, maximum, and standard deviation of speed, acceleration, and yaw rate for each MP as well as the average jerk of acceleration and of yaw rate for each MP). Data was scaled to ensure variables were equally weighted. Due to concerns about collinearity and to improve DBSCAN runtime, we first conducted a principle component analysis. We found that 8 components explained over 95% of the variation of the 14 variables. As such, we opted to set our *minPts* parameter to twice the number of components at 16. For *normPercent* and *stopThresh* we used the values of 0.9 and 3. Before each iteration was conducted, a k-nearest neighbors plot was utilized to choose that iterations ε value (which ranged from 3 – 4.5).

Overall, I-DBSCAN ran for 3 iterations when *stopThresh* was met, identifying zero abnormal clusters and a total of 87 abnormal observations identified as noise representing 56 distinct turns. The algorithm was terminated after the third iteration because a total of three iterations had occurred with only a normal cluster and noise being identified, triggering the *stopThresh* termination.

E. Examining the Results

Of the 1,231 right turn observations representing 495 distinct turns, we identified a total of 87 abnormal observations representing 56 distinct turns as noise. We identified no additional abnormal clusters. In **Table 1**, we see that the mean speed, acceleration, yaw, and jerk values of the abnormal observations are significantly different from that of the normal observations. This, though, does not yet provide us meaningful information how the noise observations differ from the normal. What is important to note is that the noise observations are not clustered together and, thus, we should not expect their variable profile to be similar. As can be seen in **Table 1**, the standard deviation of each variable (values in parentheses) is greater for the noise observations than for the normal, which is consistent with this expectation that the noise observations would be more heterogeneous than the normal observations, which represent a unified cluster.

Table 1. Comparing Average Values

	Normal (n = 1,144)	Noise (n = 87)
Average Speed (m/s)	4.70 (1.46)	5.09 (3.26)*
Average Acceleration (m/s^2)	0.32 (0.77)	0.16 (1.03)
Jerk Acceleration (m/s^3)	0.39 (0.41)	0.55 (0.71)**
Average Yaw (deg/s)	18.96 (6.81)	6.13 (20.58)***
Jerk Yaw (deg/s^2)	3.09 (6.85)	6.01 (14.59)**

*p<0.05 **p<0.01 *** p<0.001

To investigate this heterogeneity further, we can look at the extreme values of each observed variable. In **Table 2** we see the average (mean) maximum and minimum values for speed, acceleration, and yaw rate for each MP. Since the MPs are 3 seconds long (as we defined) and constructed from deci-second TP data, each MP is representative of 30 TP observations. For a given MP, the maximum speed is the TP (out of 30) which had the highest recorded speed value. **Table 2** presents the average maxima and minima for speed, acceleration, and yaw rate calculated in this way. First, we can see that the maximum speed average of noise observations is significantly greater than that of the normal observations. Then, that the absolute values of maximum and minimum accelerations of the noise observations are significantly greater than the normal observations. Finally, we see that the maximum and minimum yaw rates, similar to acceleration, are more extreme for the noise observations than they are for the normal observations.

Table 2. Comparing Extreme Values

	Normal (n = 1,144)	Noise (n = 87)
Max Speed (m/s)	6.01 (1.46)	6.63 (3.23)**
Min Speed (m/s)	3.67 (1.48)	3.79 (3.35)
Max Accel (m/s^2)	1.53 (0.88)	2.06 (1.83)***
Min Accel (m/s^2)	-0.86 (1.03)	-1.64 (1.43)***
Max Yaw (deg/s)	34.74 (11.53)	46.15 (45.96)***
Min Yaw (deg/s)	2.88 (8.23)	-36.22 (42.32)***

*p<0.05 **p<0.01 *** p<0.001

These results begin to highlight that these noise observations represent a set of turns which are, in sum, faster and with more extreme acceleration and yaw values than the normal cluster. Given, though, that the noise observations do not represent a cohesive cluster, it is necessary to examine observations individually to better confirm and interpret the results. In order to confirm that the noise observations are in fact outliers of the normal observations, plotting observations and denoting them by cluster should highlight this. Given that in **Table 2** we have identified significant differences in maximum and minimum yaw rates, we can plot observations based on their maximum and minimum yaw rates and color code them based on whether they are in the normal cluster or are noise. We see in **Figure 2** that a majority of the noise observations have (maximum yaw, minimum yaw) values which outlie those of the normal cluster. While some of the noise observations lay within the normal cluster for these two variables, repeating this plotting process with other available variables (such as maximum speed and acceleration) will similarly highlight how these observations are outliers of the normal cluster as well.

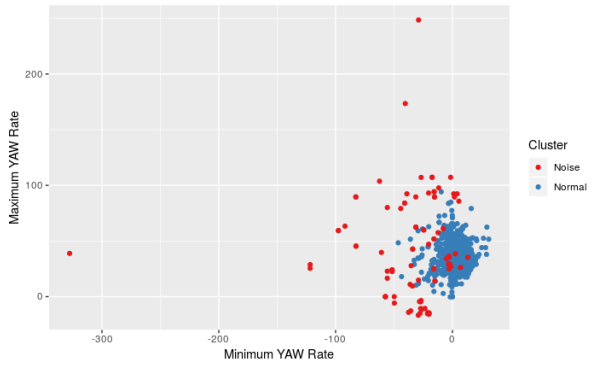

Figure 2. Max YAW vs. Min YAW

F. Interpreting the Results

The first (and most important) thing to note is that the noise observations represent a set distinct from the normal cluster (i.e. abnormal). The normal cluster can be interpreted as normal right turns. As should be noted, all of the data points examined represent moving vehicles (i.e. we have no data for stopped cars), which is because we subsetted the data based on change in heading – a stationary car cannot change heading. As the size of our data increases, by reasoning of the law of large numbers, the more right turns included in our dataset, the closer the mean should come to representing a sort of "standard" right turn [9]. We can then use this standard to compare our abnormal data and assess what makes them abnormal and if they represent aggressive driving.

The noise identified is clearly representative of outlying data. As shown in **Table 2** and **Figure 2**, the noise observations represent significantly faster turns with significantly more extreme acceleration and yaw rate values. This signals that many of these observations represent more aggressive turning behavior. This validates the application of I-DBSCAN as a protocol for identifying aggressive driving behavior from unlabeled RWD data.

G. Next Steps

It is an important, next, to discuss how the results of I-DBSCAN may be applied within a larger analytic effort utilizing RWD data to identify aggressive driving. An immediate further consideration, then, that must be made is whether there are any clusters of similar noise (i.e. abnormal clusters) that DBSCAN did not identify. One weakness of DBSCAN is that it does not identify clusters of varying density well [10]. It is possible that there are clusters amongst the noise observations that are simply less dense than the normal cluster. Regardless though, the observations identified as noise are similar in that they are outliers. These data should be understood to be abnormal and infrequent. They may represent aggressive driving behaviors which occur too rarely to be clustered together given the density of the normal cluster. Analyses can be undertaken which only examine the abnormal observations identified to see if any patterns for interpretation emerge.

Then, we can compare the results of I-DBSCAN shown here with other similar intersections. We would expect the statistical profile of right turns at similar intersections to be similar and, so, we would expect to see similar cluster structures across these intersections. As well, as noted, there may be clusters amongst the noise not being identified and increasing sample size may increase likelihood of identifying these abnormal clusters. Finally, we may expect to find differences, both in the quantity and type of abnormal behaviors identified. By comparing different intersections, we can compare where more abnormal behavior occurs and, as well, if specific kinds of abnormal behavior occur only at certain intersections but not others. This can allow us to target locations where abnormal and aggressive driving behaviors are most concentrated.

IV. Discussion

As displayed here, I-DBSCAN is an innovative tool which can be incorporated into an analytic approach for identifying abnormal and aggressive driving behaviors in unlabeled real-world driving data. I-DBSCAN, given data representing an elementary driving behavior, can identify outliers in the form of unclustered noise. Further examples should be presented to display how I-DBSCAN can identify abnormal clusters as well. These results can be utilized, among other things, to identify locations with high concentrations of abnormal driving behavior, to characterize types of abnormal driving behaviors, and to identify potential aggression driving events.

The I-DBSCAN algorithm can be viewed as a malleable strategy for identifying aggressive driving behaviors. Different datasets will consist of different variables, both in relation to vehicle movement (such as speed and acceleration), but also in relation to vehicle environment (such as road type and lane width). All such available variables should be incorporated into the procedure outlined here to enhance the ability to both identify EDBs and to then implement I-DBSCAN.

Characterizing unlabeled data remains a great challenge in efforts to identify aggressive driving behaviors from RWD data. As displayed in our case example, I-DBSCAN has the capacity to identify abnormal and aggressive driving behaviors from unlabeled RWD data. The results of I-DBSCAN can be utilized to compare the rate of and the form of abnormal driving behaviors across comparable environments (for example, at similar intersections) for the purpose of identifying locations where abnormal, and potentially aggressive, driving behaviors are concentrated. This can help agencies such as cities to identify infrastructure failures (e.g. potholes) and possible design issues (e.g. geometric design, signal timing).

While the core method of I-DBSCAN is simply the repetitive application of DBSCAN to subsets of RWD data with a set of restrictions, the innovation of this technique lies in the full process defined. The steps defined in implementing I-DBSCAN represent a start-to-finish process of taking raw RWD data, preparing it for analysis, and identifying aggressive driving data points.

In order to optimize the utilization of I-DBSCAN, implementation must be made simple with easily interpretable results. The development of I-DBSCAN statistical tools (such as R libraries and GUI applications) can reduce barriers to implementation and simplify the process of undertaking the data formatting and analyses described here. In addition, further research should be undertaken to better define the process of interpreting the results of I-DBSCAN and in order to recommend methods for further analyzing the abnormal observations identified by the algorithm. As well, sensitivity analyses should be undertaken to determine if variations in I-DBSCAN implementation, such as changes to the length of MP data, change the performance of the algorithm overall. Finally, as is a danger with many machine learning techniques, analyses should be undertaken to determine model overfitting.

V. Conclusion

Every year, millions are injured or killed due to aggressive driving across the globe. I-DBSCAN can be utilized by organizations seeking to reduce the harms of traffic accidents by identifying abnormal driving behaviors. This can help with prioritizing locations where aggressive driving is observed more frequently and developing appropriate countermeasures. Future efforts should focus on defining a protocol by which the results of I-DBSCAN can trigger actions which will actively improve traffic safety by highlighting environments where abnormal, potentially aggressive, driving behaviors are most concentrated.

Acknowledgment

This research was supported by Safe-D University Transportation Center. The contents of this paper reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This work is funded by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

VI. References

[1] World Health Organization, "Global status report on road safety 2018," World Health Organization, Geneva, 2018.

[2] AAA Foundation for Traffic Safety, "Aggressive Driving: Research Update," AAA , Washington DC, 2009.

[3] M. Ester, H.-P. Kriegel, J. Sander and X. Xiaowei, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases With Nose," in *2nd International Conference on Knowledge Discovery and Data Mining*, Portland, 1996.

[4] K. Beyer, J. Goldstein, R. Ramakrishnan and U. Shaft, "When Is "Nearest Neighbor" Meaningful?," in *International Conference on Database Theory*, Jerusalem, 1999.

[5] J. Sander, M. Ester, H.-P. Kriegel and X. Xu, "Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications," *Data Mining and Knowledge Discovery,* vol. 2, no. 2, pp. 169-194, 1998.

[6] E. Schubert, J. Sander, M. Ester, H.-P. Kriegel and X. Xu, "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN," *ACM Transactions on Database Systems,* vol. 42, no. 3, 2017.

[7] D. Bezzina and J. Sayer, "Safety Pilot Model Deployment Test Conductor Team Report," National Highway Traffic Safety Administration. , Washington DC, 2015.

[8] Intelligent Transportation Systems Join Program Office, "RESEARCH DATA EXCHANGE RELEASE 2.3 SAFETY PILOT MODEL DEPLOYMENT DATA," US Department of Transportation.

[9] Encyclopedia of Mathematics, "Law of Large Numbers," Encyclopedia of Mathematics, 13 5 2012. [Online]. Available: https://www.encyclopediaofmath.org/index.php/Law_of_large_numbers. [Accessed 22 3 2019].

[10] H.-P. Kriegel, P. Kroger, J. Sander and A. Zimek, "Density-based clustering," *WIREs Data Mining and Knowledge Discovery,* vol. 1, no. 3, pp. 231-240, 2011.