**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 24, No. 1, October 2021, pp. 444~451
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v24.i1.pp444-451


# Incident forecasting model for motorcycle driving based on IoT and artificial intelligence

**Esteban Alejandro Cárdenas-Lancheros[1], Nelson Enrique Vera-Parra[2]**
[1,2]Faculty of Engineering, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia
[2]GICOGE Research group, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia

**Article Info**

***Article history:***

Received Jun 3, 2021
Revised Aug 4, 2021
Accepted Aug 7, 2021

***Keywords:***

Accelerometer
Artificial intelligence
Driving behavior
Internet of things
Machine learning
Motorcycle

**ABSTRACT**

Internet of things (IoT) and artificial intelligence provide more and more solutions to the exercise of capturing data effectively, taking them through processing and analysis stages to extract valuable information. Currently, technological tools are applied to counteract incidents in motorcycle driving, whether they are part of the same vehicle or are externally involved in the environment. Incidents in motorcycle driving are increasing due to the demand for the acquisition of these vehicles, which makes it important to generate an approach towards reducing the risk of road accidents based on the analysis of dynamic behavior while driving. The development of this research began with the detection and storage of data associated with the dynamic acceleration variable of a motorcycle while driving, this with the help of a 3-axis accelerometer sensor generating a dataset, which was processed and analyzed for later be taken by three predictive classification models based on machine learning which were decision trees, K-Nearest neighbors and random forests. The performance of each model was evaluated in the task of better classifying the level of accident risk, concerning the driving style based on certain levels of acceleration. The random forest model showed a slightly better performance compared to that shown by the other two models, with 97.24% accuracy and recall, 97.16% precision and 97.17% F1 score.



***Corresponding Author:***

Esteban Alejandro Cárdenas Lancheros
Faculty of Engineering
Universidad Distrital Francisco José de Caldas
Bogotá, Colombia
Email: clestebana@correo.udistrital.edu.co

## 1. INTRODUCTION

Cities, having a road infrastructure, can show a variable circulation of motorcycles. In the case of Bogotá-Colombia, from January to December 2020, 518,666 motorcycles were registered, with a growth of 4.14% compared to the same period in 2019 [1]. The use of the motorcycle becomes an alternative that presents certain advantages compared to transport by car. The main advantages are associated with money and time, since fuel and maintenance costs are lower; in terms of time, transport by motorcycle in the city is faster, making the same trip that makes a car by 50% or 70% less time.

However, it has a disadvantage related to a higher risk of accidents [2]. Increases in motorcycle sales in recent years reflect an increase in the numbers of deaths and serious injuries caused to their drivers [3], [4]. Motorcycle users represent 53.2% of the total number of deaths and 59.7% of the total number of injuries registered in the country due to road accidents [5], the high speeds, the instability of the vehicle and the

inadequate state of the road, can cause an accident that will directly affect the integrity of the driver because he is exposed and is only protected by gloves, boots, helmet and clothing in general.

With driver behavior and congested road conditions, accidents often occur. Applications developed for smartphones can generate alerts to avoid accidents by analyzing data from the accelerometer and gyroscope sensors incorporated in the phone. Research has adopted machine learning-based motion identification processes. The device calculates activity based on several predetermined categories of driver status. Status categories can be normal, zigzag, sleepy, right turn, left turn, U-turn, sudden braking, and sudden acceleration [2].

Machine learning can be very useful in recognizing driving patterns. The problem is formulated as a classification task to identify the class of driving patterns using data collected from the sensors. Several machine learning techniques can be implemented, including Gaussian mixture models, the k-nearest neighbor model, decision trees, support vector machines, random forests, and hidden Markov models (HMM), where these last two show a high performance in the recognition of driving patterns [6], [7].

Each motorcycle is taken as an individual agent and its dynamic acceleration variable is studied, which is present during the time that the activity elapses [8] and is captured by a sensor in the longitudinal and lateral axes [9], this data goes through a transformation phase to extract interpretable statistical characteristics from this raw data [10], resulting in the creation of a training dataset which can be used in the investigation of certain events or decisions [11]. A classification into four categories associated with levels of risk in driving will predict new data from the training dataset and give the answer to the correct class [12], [13]. In this research, three machine learning classifier models will be taken, they will be compared, and evaluated, to validate which is the one that shows the best performance to classify levels of accident risk, thus leading to the forecast of incidents in driving.

## 2. RESEARCH METHOD

### 2.1. Dataset creation

From a data article entitled "Dataset on powered two wheelers fall and critical events detection" [14], Several sub-datasets are extracted, which will be transformed, analyzed and unified, in order to be able to generate a predictive classification model that works with these data and leads to the forecasting of motorcycle driving incidents. For the development of this research, the data will be taken from the mentioned data article. The data article presents the data generated by a 3D inertial measurement unit, which incorporates 3 accelerometers and 3 gyroscopes. For this investigation only the acceleration components $(a_x, a_y, a_z)$ will be used. The inertial measurement unit was mounted on the motorcycle collecting the data during 8 controlled experiments, capturing every second a sample of the magnitude of acceleration for each of the three spatial dimensions $(a_x, a_y, a_z)$ [15], the sample will be stored in a register. The magnitude of acceleration is given in $m/s^2$.

#### 2.1.1. Dataset transformation

The aim is to extract from the sub-datasets, metrics based on exploratory statistical measures, to be able to make interpretations about the data [16]. This requires dividing each sub-dataset into windows and sub-windows of tuples, so that exploratory statistics measurements can be performed on each of these, for $a_x, a_y, a_z$ respectively. Each sub-dataset contains between 50,000 and 130,000 tuples, windows of 100 records were established on each sub-dataset, likewise, sub-windows were generated which are in the middle of two windows also of 100 tuples. Windows and sub-windows will become future new tuples. A value of 100 tuples was chosen for each window and sub-window, to maintain a large set of new tuples. Sample size is an important consideration for research. Larger sample sizes provide more accurate mean values, identify outliers that could skew the data in a smaller sample and provide a smaller margin of error [17].

A sub-window, which contains 100 tuples, is in the middle of two windows, that is, it covers the last 50 tuples of the first window and the first 50 tuples of the second window, the second sub-window covers the last 50 tuples of the second window and the first 50 tuples of the third window, and so on generating an overlapping set of new tuples. Metrics based on exploratory statistical measures will be extracted from this overlapping set. The metrics will be based on the following types of exploratory statistics measures: i) Measure of central tendency: average; ii) Position measurements: maximum, minimum Q1 quartile (25%), Q2 quartile (50%) and Q3 quartile (75%); and iii) Dispersion measures: variance and standard deviation.

There is a total of 8 metrics based on exploratory statistics measures, these 8 metrics will be calculated for each of the 3 magnitudes of acceleration $a_x, a_y, a_z$ in each sub-window, obtaining a total of 24 metrics or input variables for each sub-window. There will be 8 new transformed datasets corresponding to the 8 controlled experiments, these will initially have 24 columns corresponding to each metric that here will also be called the input variable, by $n$ number of sub-windows (tuples). The values of each of the 24 metrics

must be plotted, their behavior analyzed and a conclusion will be obtained per sub-window, which will lead to generate a label for that sub-window, located in a last column (number 25) of the new transformed sub-dataset, which will be taken as the output variable [18], [19]. The 8 sub-datasets must be joined and form a final dataset, this will be hosted in a GitHub repository. This final data set is taken by a machine learning model, which consists of an algorithm trained by these data; the model will provide an output which is considered a forecast based on the data that trained the model [20]-[23].

### 2.1.2. Data analyst

When manual labeling is performed for each sub-window, this labeling will correspond to a certain category of risk level in motorcycle driving. The definition of the categories is based on longitudinal and lateral acceleration levels $(a_x, a_y)$ associated with dangerousness [24] and with particular ways of driving, such as a professional driver would do in a daring and sporty way or even as a beginner would drive in a more calm and relaxed way [25]. The manual labeling procedure is based on the assignment of a risk level for each sub-window, there are 4 risk level options, and each level will have associated a numerical value as shown in Table 1. Table 1, in the last column, which is number 25, will be the data of the numerical value associated with each risk level. The following criteria are taken in account to assign a risk level to each sub-window, these will be considered only for the variables related to the components of acceleration for the "x" and "y" axes $(a_x, a_y)$, since on the z axis $(a_z)$ the acceleration component of the earth's gravity force is manifested, and it tends to remain constant [26], [27]. These criteria are mentioned:

- Average that fits within each acceleration range from Table 1: Being the mean of the data in each sub-window, the definition of its central tendency is an important measure to consider.
- Maximum and minimum not too far apart: The fact that these two measurements are not too far apart shows that there is greater uniformity in the value of the data in each sub-window.
- Quartiles Q1 and Q2 that fit or are close to each acceleration range in Table 1: Having in 25% and 50% of the observations data that fit within a range of accelerations significantly determines in which category of risk level the sub-window would be.
- Variance and standard deviation not very high: These two measures indicate how dispersed the values of the sample data can be, for this case, if the data is more concentrated, they show a similar trend.

It is enough that one of the two acceleration components, either $a_x$ o $a_y$ shows a noticeable change to categorize the sub-window, otherwise it will remain in the "Low" category with a numerical value of "0". In Figure 1, the data behavior for a sub-window is shown, we see that for the component $a_x$, the mean, the distance between the maximum and the minimum, the quartiles Q1 and Q2, the variance and the standard deviation conform to the aforementioned criteria and allow us to decide that the most appropriate category for this sub-window is a "Notable" risk level with a numerical value of "1".

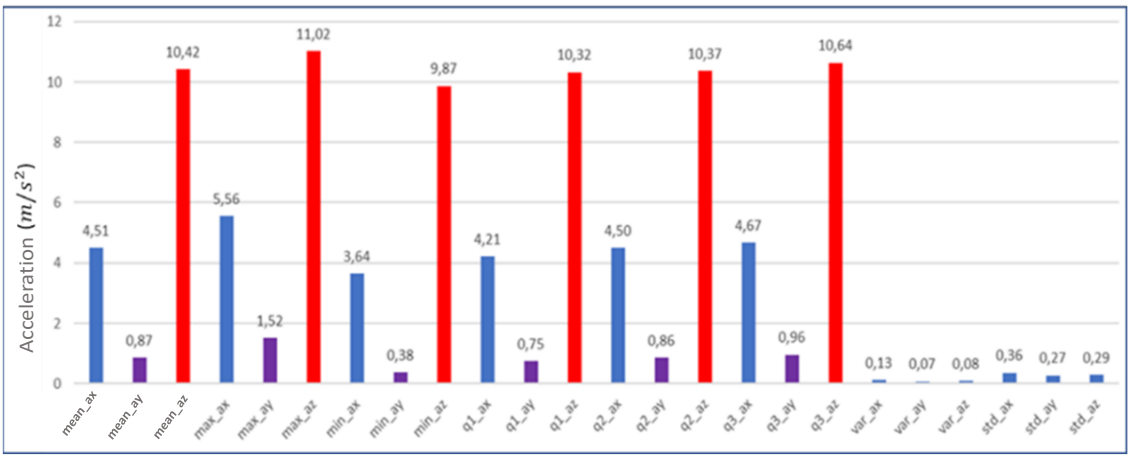


Figure 1. Behavior of the 24 metrics corresponding to a sub-window

Table 1. Risk levels and their associated numerical values

| Risk level | Acceleration range $(m/s^2)$ | Numerical value |
|---|---|---|
| Low | 0 – 2,5 | 0 |
| Notable | 2,5 – 5 | 1 |
| High | 5 – 7 | 2 |
| Very high | > 7 | 3 |

### 2.2. Proposed machine learning models

There is a dataset with 25 columns, where 24 of them correspond to the respective metrics or input variables obtained from the exploratory statistics measures, and column number 25 or output variable, would be the risk level column corresponding to the manually labeled data or class. Three widely used predictive models of machine learning based on supervised learning are proposed, which work with classification algorithms and have presented good results for problems of a similar nature [28]-[30]; these are mentioned:

Decision trees (DT): decision trees are sequential models, logically combining a sequence of simple tests; each test compares a numeric attribute to a threshold value or a nominal attribute to a set of possible values. When a data point falls into a partitioned region, a decision tree classifies it as belonging to the most frequent class in that region [31].

KNN (k-nearest neighbors): This method finds the k closest neighbors of the labeled instances to the unlabeled instance using the Euclidean distance between the feature vectors. Returns the label that represents the most neighbors [32]. Random forests (RF): random forests are a combination of predictor trees in such a way that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. Classification problems are solved by analyzing the output of the trees. Most of the votes of the class or category determines the prediction of the random forests [33].

#### 2.2.1. Implementation of the models

The python programming language is highly suitable in terms of model implementation, as it is establishing itself as one of the most popular languages for scientific computing. Thanks to its high-level interactive nature and scientific library ecosystem, it is an attractive option for algorithmic development and exploratory data analysis [34], [35]. Prior to the implementation of each of the three machine learning models, it is necessary to properly divide the final dataset into training and testing subsets. The training subset is applied to train or adjust the model and the test subset is necessary for an evaluation of the final model [36]. Additionally, proper splitting minimizes the potential for bias in the evaluation and validation process [37]. Of the total data, 70% is designated as a training subset and 30% as a test subset.

## 3. RESULTS AND DISCUSSION

### 3.1. Comparison of machine learning models

After implementing the three machine learning models, we proceed to validate how effective the capacity of each proposed model is to correctly identify classes. Validation will be based on a confusion matrix as shown in Figure 2 which will define the following metrics: accuracy, precision, recall and F1 score [38].

For the accuracy metric, in addition to exposing the results of each of the three models proposed in this investigation, a comparison will be made of these results with those generated by two other previous investigations; the first proposes a machine learning framework for driving pattern recognition [6] and will be named "Investigation 1", the second focuses on detecting driving events by applying the method of learning sets for classification [7], we will name this "Investigation 2". The results of the accuracy metric for this investigation, “Investigation 1” and “Investigation 2”; are observed in Table 2.

| | | | |
|---|---|---|---|
| 683 | 13 | 0 | 0 |
| 11 | 191 | 8 | 0 |
| 0 | 4 | 41 | 2 |
| 0 | 0 | 2 | 16 |

(a)

| | | | |
|---|---|---|---|
| 690 | 6 | 0 | 0 |
| 16 | 187 | 7 | 0 |
| 0 | 10 | 33 | 4 |
| 0 | 0 | 2 | 16 |

(b)

| | | | |
|---|---|---|---|
| 695 | 1 | 0 | 0 |
| 16 | 188 | 5 | 1 |
| 0 | 7 | 36 | 4 |
| 0 | 1 | 3 | 14 |

(c)

Figure 2. Confusion matrices: (a) decision trees, (b) KNN and (c) Random forests

Table 2. Accuracy metric

| Model | Accuracy | | |
|---|---|---|---|
| | This investigation | Investigation 1 | Investigation 2 |
| Decision trees | 95,9% | - | 84,74% |
| K - nearest neighbors | 95,4% | 82,4 ± 7,4 % | 85,18% |
| Random forests | 96,1% | 84,7 ± 7,6 % | 94,07% |

The accuracy percentages of the three models proposed in this investigation are somewhat higher than those that resulted from investigations 1 and 2. The three investigations attempt to solve problems of

similar natures; however, their methodologies may differ; it is observed that the random forest model for the three scenarios presents a higher accuracy. With the 'weighted' average parameter, the other metrics are calculated for each label. By having a class imbalance, which occurs when there is a significant difference in the amount of data corresponding to each class [39]; a weighted average makes more sense, where the weights are calculated by the frequency of a certain class, weighting the metric of each class by the number of samples of that class [40].

$$Pr_{Weighted} = Pr_1 \frac{\#Obs_1}{N} + Pr_2 \frac{\#Obs_2}{N} + \cdots + Pr_k \frac{\#Obs_k}{N} \quad (1)$$

Results-metrics with a 'weighted' average parameter of precision, recall and F1 score are observed in Table 3 and in Figure 3. The three models show close performance ratios; however, the random forest model has the highest performance ratio; according to this, it is the model that best suits the investigation.

Table 3. Precision, recall and F1 score metrics; with 'weighted' average parameter

| Model ('weighted' average) | Precision | Recall | F1 score |
|---|---|---|---|
| Decision trees | 96,1% | 96,0% | 96,0% |
| K - nearest neighbors | 95,1% | 95,2% | 95,0% |
| Random forests | 96,2% | 96,3% | 96,2% |

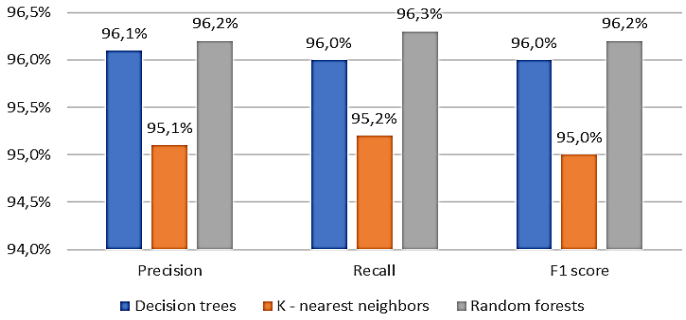


Figure 3. Precision recall and F1 score metric, with 'weighted' average parameter

### 3.2. Results of the evaluation metrics for the chosen model with adjusted hyperparameters

Result-accuracy metric is observed in Table 4. To obtain precision, recall and F1 score, the 'weighted' average parameter was considered, this result is observed in Table 5. With the adjustment of hyperparameters, an improvement in the performance of the model was generated, this is evidenced in Table 6 and in Figure 4. The adjusted hyperparameters were "n_estimators"=100 and "random_state"=100.

Table 4. Accuracy metric-random forests model with adjusted hyperparameters

| Model | Accuracy |
|---|---|
| Random Forests | 97,24% |

Table 5. Precision, recall and F1 score metrics; with 'weighted' average parameter-random forests model with adjusted hyperparameters

| Model ('weighted' average) | Precision | Recall | F1 score |
|---|---|---|---|
| Random Forests | 97,16% | 97,24% | 97,17% |

Table 6. Improved performance of the random forest model due to the adjustment of hyperparameters

| | Adjustment of hyperparameters | | |
|---|---|---|---|
| Evaluation metric | Without | With | Improvement |
| Accuracy | 96,10% | 97,24% | 1,19% |
| Precision | 96,20% | 97,16% | 1,00% |
| Recall | 96,30% | 97,24% | 0,98% |
| F1 score | 96,20% | 97,17% | 1,01% |

Apart from the hyperparameters that are established by default, the number of estimators "n_estimators" is kept at the maximum (100), thus there will be a greater number of trees which increases the

performance making the predictions more stable; additionally, a parameter to adjust is the "random-state", which controls both the randomness of the samples used when constructing trees and the sampling of the characteristics to consider when looking for the best division in each node, it is adjusted to 100, thus obtaining the best result following a deterministic behavior. Importance measures of variables produced by random forests have also been suggested for the selection of relevant predictor variables in data analysis, however, these variable importance measures show a bias towards correlated predictor variables [41]. This measure is included and shown in Figure 5.

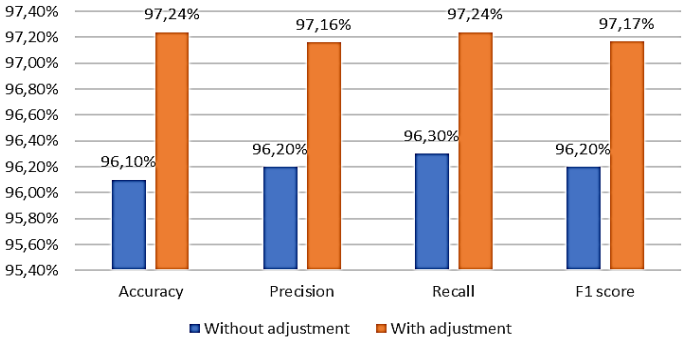


Figure 4. Improved performance of the random forest model due to the adjustment of hyperparameters.

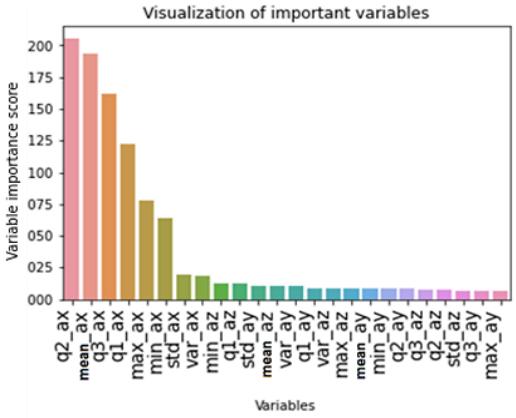


Figure 5. Variables score by degree of importance

## 4. CONCLUSION

According to the measurement of importance of variables, those with the highest score are the Q2 quartile, followed by the mean, the Q3 quartile, the Q1 quartile, maximum and minimum; they are associated with the x axis (longitudinal acceleration), since it is in this axis where a greater behavior is evidenced within the dataset. It would be expected that, if there were a greater activity in the y-axis (lateral acceleration), a similar situation would be reflected here in terms of the score of the importance of variables. Since the output variable of the risk level results in an attribute of 4 possible classes, the evaluation of all the classes must be considered, to add them and obtain the actual evaluation of the classifier model for the attribute. Here, formulating this, we have introduced the evaluation of classification weighted average for the general recognition rate of the classifier model, which reflects how well the classifier model recognizes the attributes of the 4 classes. The three models implemented resulted in good indexes in terms of class prediction, however, the Random Forests model has a slight advantage in the results generated by its evaluation metrics compared to the Decision Trees (average of 0.21% higher) and K nearest neighbors (average 1.29% higher) models, due to its simplicity and randomization process. Random forests are considered an effective prediction tool and by adjusting their hyperparameters, an average improvement of close to 1.04% of the results generated by their evaluation metrics can be achieved. As a future development, it could generate our own dataset where there is greater activity related to lateral accelerations and validate the importance within the variables that characterize the risk of danger, likewise, be able to consider the variable from the gyroscope (angular velocity in the three spatial axes), performing a similar data processing and analysis, thus adding one more characteristic to the forecast of incidents in motorcycle driving. Other predictive

classification models based on supervised learning can be implemented, such as bayesian networks, neural networks or support vector machines, and with this validate the performance that they could have in a problem of this nature.

## REFERENCES


[1] A. Fenalco, "Informe De Matrículas De Motos A Diciembre De 2020," *Bogotá*, 2021.

[2] F. M. Nuswantoro, A. Sudarsono, and T. B. Santoso, "Abnormal Driving Detection based on Accelerometer and gyroscope sensor on smartphone using Artificial Neural Network (ANN) algorithm," *International Electronics Symposium (IES),* pp. 356-363, 2020, doi: 10.1109/IES50839.2020.9231851.

[3] D. Clarke, P. Ward, C. Bartle, and W. Truman, "In-depth Study of Motorcycle Accidents," *Road Safety Research Rep*, vol. 54, 2004.

[4] B. Pharmacopoeia, "Controller of Her Majesty's Stationery Office," *Norwish*, vol. 1, p. 805, 2004.

[5] Observatorio Nacional de Seguridad Víal, "Agencia Nacional de Seguridad Víal," 2021. Accessed: April 8, 2021. [Online]. Available: https://ansv.gov.co/observatorio

[6] F. Attal, A. Boubezoul, L. Oukhellou, and S. Espié, "Powered Two-Wheeler Riding Pattern Recognition Using a Machine-Learning Framework," *in IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 475-487, Feb. 2015, doi: 10.1109/TITS.2014.2346243.

[7] B. Bose, J. Dutta, S. Ghosh, P. Pramanick, and S. Roy, "Smartphone based system for real-time aggressive driving detection and marking rash driving-prone areas," *Association for Computing Machinery - International Conference Proceeding Series,* no. 27, pp. 1-6, 2018, doi: 10.1145/3170521.3170549.

[8] I. Zafar and K. M. Iqbal, "Automatic incident detection in smart city using multiple traffic flow parameters via V2X communication," *International Journal of Distributed Sensor Networks,* vol. 14, no. 11, pp. 1-23, 2018, doi: 10.1177/1550147718815845.

[9] F. Attal, A. Boubezoul, L. Oukhellou, and S. Espié, "Analysis and classification of Powered Two Wheelers Riding Pattern," *Transport Research Arena (TRA) 5th Conference: Transport Solutions from Research to Deployment,* no. 01540761, 2014, pp. 1-10.

[10] Z. Li, K. Zhang, B. Chen, Y. Dong, and L. Zhang, "Driver identification in intelligent vehicle systems using machine learning algorithms," *IET Intelligent Transport Systems,* vol. 13, no. 1, pp. 40-47, 2018, doi: 10.1049/iet-its.2017.0254.

[11] E. Ahmed, I. Yaqoob, I. A. Targio Hashem, I. Khan, A. M. I. Abdalla Ahmed, M. Imran, and A. V. Vasilakos, "The role of big data analytics in Internet of Things," *Computer Networks,* no. S1389-1286(17)30259-1, p. 1, 2017, doi: 10.1016/j.comnet.2017.06.013.

[12] A. H. Shaon, "Identification of Dangerous Driving Behavior Using Naturalistic Driving Data," *Munich: Technical University of Munich*, 2019.

[13] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques," *University of Peloponnese, Greece,* pp. 248-268, 2007.

[14] A. Boubezoul, F. Dufour, S. Bouaziz, B. Larnaudie, and S. Espié, "Dataset on powered two wheelers fall and critical events detection," *ELSEVIER,* vol. 23, no. 2352-3409, pp. 1-6, 2019, doi: 10.1016/j.dib.2019.103828.

[15] A. Boubezoul, S. Espié, BrunoLarnaudie, and S. Bouaziz, "A simple fall detection algorithm for powered two wheelers," *ELSEVIER,* vol. 21, no. 3, pp. 286-297, 2013, doi: 10.1016/j.conengprac.2012.10.009.

[16] K. Sahoo, A. K. Samal, J. Pramanik and S. K. Pani, "Exploratory Data Analysis using Python," *International Journal of Innovative Technology and Exploring Engineering,* vol. 8, no. 2278-3075, pp. 4727-4729, 2019.

[17] J. Zamboni, "Sciencing," 2018. Accessed: Apr. 20, 2018]. [Online]. Available: https://sciencing.com/advantages-large-sample-size-7210190.html

[18] T. Hastie, R. Tibshirani, and J. Friedman, "Overview of Supervised Learning," in *The Elements of Statistical Learning Data Mining, Inference, and Prediction*, Stanford CA, USA, Springer, pp. 9-11, 2009, doi: 10.1007/b94608_2.

[19] A. E. Mohamed, "Comparative Study of Four Supervised Machine Learning Techniques for Classification," *International Journal of Applied Science and Technology,* vol. 7, no. 2, pp. 5-18, 2017.

[20] P. Cunningham, M. Cord, and S. J. Delany, "Supervised Learning," in *Machine Learning Techniques for Multimedia*, Berlin, Springer, pp. 21-22, 2008.

[21] H. Bhavsar and A. Ganatra, "A Comparative Study of Training Algorithms for Supervised Machine Learning," *International Journal of Soft Computing and Engineering,* vol. 2, no. 4, pp. 74-81, 2012.

[22] N. Raju, S. S. Arkatkar and G. Joshi, "Experiences of supervised Machine Learning to replicate driving behavior under traffic environment," *Conference paper - researchgate,* 2019, p. 16.

[23] N. M. N. Mathivanan, N. A. Md.Ghani, and R. M. Janor, "Performance analysis of supervised learning models for product title classification," *IAES International Journal of Artificial Intelligence (IJAI),* vol. 8, no. 3, pp. 299-306, 2019, doi: 10.11591/ijai.v8.i3.pp228-236.

[24] Fu Li, Hai Zhang, Huan Che, and Xiaochen Qiu, "Dangerous driving behavior detection using smartphone sensors," 2016 *IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 1902-1907, doi: 10.1109/ITSC.2016.7795864.

[25] S. Will, B. Metz, T. Hammer, M. Mörbe, M. Henzler, F. Harnischmacher, and G. Matschl, "Methodological considerations regarding motorcycle naturalistic riding investigations based on the use of g-g diagrams for rider profile detection," *ELSEVIER,* pp. 1-12, 2020, doi: 10.1016/j.ssci.2020.104840.

[26] P. N. Morales, "Sensors - Sensors Offset," in *Smart Movement Detection For Android Phones*, Barcelona, Universitat Politècnica de Catalunya, 2016, pp. 15-23.
[27] M. Tecpoyotl-Torres, R. Cabello-Ruiz, P. Vargas-Chable, J. G. Vera-Dimas, and A. Ocampo-Diaz, "Performance of compliant mechanisms applied to a modified shape accelerometer of single and double layer," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 9, no. 6, pp. 4675-4683, 2019, doi: 10.11591/ijece.v9i6.pp4675-4683.
[28] N. Dogru and A. Subasi, "Traffic accident detection using random forest classifier," 2018 *15th Learning and Technology Conference (L&T)*, 2018, pp. 40-45, doi: 10.1109/LT.2018.8368509.
[29] M. Rezapour, A. M. Molan, and K. Ksaibati, "Analyzing injury severity of motorcycle at-fault crashes using machine learning techniques, decision tree and logistic regression models," *International Journal of Transportation Science and Technology,* vol. 9, no. 2, pp. 89-99, 2020, doi: 10.1016/j.ijtst.2019.10.002.
[30] B. K. Mohanta, D. Jena, N. Mohapatra, S. Ramasubbareddy, and B. S. Rawal, "Machine learning based accident prediction in secure IoT enable transportation system," *Journal of Intelligent & Fuzzy Systems,* pp. 1-13, 2021.
[31] S. B. Kotsiantis, "Decision trees: a recent overview," *Springer,* vol. 38, no. 4, pp. 261-262, 2011, doi: 10.1007/s10462-011-9272-4.
[32] A. D. McDonald, J. D. Lee, C. Schwarz, and T. L. Brown, "Steering in a Random Forest: Ensemble Learning for Detecting Drowsiness-Related Lane Departures," *Human Factors and Ergonomics Society.,* vol. 56, no. 5, pp. 986-998, 2013, doi: 10.1177/0018720813515272.
[33] L. Breiman, "Random Forests," *Machine Learning - Springer,* vol. 45, no. 1, pp. 5-6, 2001.
[34] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, and M. Blondel, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.
[35] Y. Ren, "Python Machine Learning - Book Review," *International Journal of Knowledge-Based Organizations,* vol. 11, no. 1, pp. 67-70, 2021.
[36] K. Malhotra and A. P. Singh, "Implementation of decision tree algorithm on FPGA devices," *IAES International Journal of Artificial Intelligence,* vol. 10, no. 1, pp. 131-138, 2021, doi: 10.11591/ijai.v10.i1.pp131-138.
[37] D. Kang and S. Oh, "Balanced training/test set sampling for proper evaluation of classification models," *Intelligent Data Analysis,* vol. 24, no. 1, pp. 5-18, 2020, doi: 10.3233/IDA-194477.
[38] D. Sarkar, R. Bali and T. Sharma, "Building, Tuning, and Deploying Models," in *Practical machine learning with Python*, Bangalore, Apress, pp. 272-275, 2018, doi: 10.1007/978-1-4842-3207-1_5.
[39] C. Kim, H. Lee, and H. Jung, "Fruit tree disease classification system using generative adversarial networks," *International Journal of Electrical and Computer Engineering,* vol. 11, no. 3, pp. 2508-2515, 2021, doi: 10.11591/ijece.v11i3.pp2508-2515.
[40] V. Patro and M. R. Patra, "Augmenting Weighted Average with Confusion Matrix to Enhance Classification Accuracy," *Transactions on Machine Learning and Artificial Intelligence,* vol. 2, no. 4, pp. 77-90, 2014, doi: 10.14738/tmlai.24.328.
[41] C. Strobl, A. L. Boulesteix, T. A. Thomas Kneib, and A. Zeileis, "Conditional variable importance for random forests," *BMC Bioinformatics,* vol. 9, no. 307, pp. 1-11, 2008, doi: 10.1186/1471-2105-9-307.

## BIOGRAPHIES OF AUTHORS

**Esteban Alejandro Cárdenas-Lancheros** is a student of the master's program in information and communication sciences at the Universidad Distrital Francisco José de Caldas (Bogotá, Colombia), Electronic Engineer from the same University.

**Nelson Enrique Vera-Parra** is a Professor and Coordinator of the master's program in information and communication sciences at the Universidad Distrital Francisco José de Caldas (Bogotá, Colombia), Doctor of Engineering from the same University, Electronic Engineer from the Universidad Surcolombiana (Neiva, Colombia), Researcher in parallel computing, high performance computing, science data and bioinformatics.