Development Of A Predictive Model Of Automobile Accidents In The City Of Bogotá, Colombia

Fredys A. Simanca H.[1], Fabian Blanco Garrido[2], Lugo Manuel Barbosa Guerrero[3], Alexandra Abuchar Porras[4], Jairo Jamith Palacios Rozo[5]

1 Profesor Investigador, Universidad Cooperativa de Colombia, Bogotá – Colombia, email: fredys.simanca@campusucc.edu.co
2 Profesor Investigador, Universidad Cooperativa de Colombia, Bogotá – Colombia, email: fabian.blanco@campusucc.edu.co
3 Docente de Planta, Colegio Mayor de Cundinamarca, Bogotá – Colombia, email: lmbarbosa@unicolmayor.edu.co
4 Docente de Planta, Universidad Distrital Francisco José de Caldas, Bogotá – Colombia, email: aabucharp@udistrital.edu.co
5 Docente de Planta, Colegio Mayor de Cundinamarca, Bogotá – Colombia, email: jjpalacios@unicolmayor.edu.co

Summary:

The importance of information analysis has allowed great advances in recent years, including predictive analysis, which, based on a series of data, look for patterns to forecast unknown data. For this article, one of the algorithms for predictions within Machine Learning is used, such as Random Forest. And, the algorithm is made in the Python programming language, in order to obtain the possible causes of accidents in the city of Bogotá. This information is taken from the open data of accidents registered in the city in order to know how many accidents will exist each day and what will be the causes to be analyzed and thus, find a concrete solution to these problems of accident rate with the purpose of reducing them considerably. In conclusion, results were obtained from the development of the algorithm that allow to predict with a certain degree of accuracy the automobile accident rate in the city.

Keywords: Accidents, Random Forest, accident prediction, accident prediction.

1. Introduction

Predictive analysis as stated in , is one where a large number of techniques for statistics and machine learning are grouped in relation to the patterns that are extracted from the relevant data to disclose unknown events. Initially, it seeks to find the relationships between the variables that are presented so that a logical connection can be found so that the values that are not yet known can be identified. However, the reliability of the values will depend on the level of depth of the analysis.[1]

Likewise, it speaks that this method expects a specific numerical value, for this, two stages are conceived that are:[2]

- Training stage
- Testing stage

[2]As he states, the first phase presents a large amount of data which is separated to train the algorithm and provide all the information to find the necessary patterns. Then, the remaining data is used to make tests, where it is questioned if the answers obtained are correct or not, for this, a margin of confidence that exceeds 80% is handled to consider that there is a high degree of learning by the algorithm.

Likewise, predictive analysis allows to give a great perspective of the future due to the great probability that it can provide by applying analysis techniques to the data. These algorithms can be used to predict demand, bank fraud, consumer behavior, identify risks, among others, which, as mentioned in , will make many companies and organizations see predictive analysis as an indispensable tool that will allow them to increase the confidence of the strategies or decisions that will be taken based on what is predicted. [3] [4] [5] [6]

It should be noted that as mentioned in , this area of data mining having the great capacity to store and process data sets, allows to extract information and behavior patterns accurately, which makes people are seen more as a data provider, this has led

to the generation of huge files of information and, Therefore, the increase of predictive analysis as a tool that allows analyzing the data that are suitable to look for this relationship between variables.[4]

On the other hand, in Colombia the use of technologies has led to an exponential advance in the management of information in the agro-industrial sectors, food, commerce, education, government, etc. However, it is mentioned that only 32% take advantage of analytics to innovate and develop businesses, 39% usually do not and 11% never apply any data analysis process, specifically because they do not have inputs for data or lack technological tools; However, little by little we begin to see the use of this tool from the smallest foundations, so that these prediction algorithms either benefit several communities and sectors such as Industry 4.0 or the different sectors of the country. For this case, it focused on the mobility sector and, therefore, the following case is presented: [6][5]

It is important to highlight that this sector is taken given the difficulties that arise, in it is said that the accident figures in Colombia are serious, being even a serious health problem, since they are the second cause of death after homicides, with 7,000 deaths in road accidents and 35,000 injured, then, This problem is reinforced with the economic impact it generates, being around 2.9 trillion pesos. [6]

Likewise, when focusing on Bogotá, the following figures are presented:

Table 1. History of injured and deceased victims from 2011 to 2020 in the city of Bogotá D.C, Taken from .[7]

	2011	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**
Bogota	489	522	476	583	516	545	505	474	455	350

Table 1 shows the total number of victims in the country's capital, showing the high accident rate for several years, handling an average of 490 accidents and only showing the highest value in 2014 with 583 accidents and the lowest value in 2020, the year when the COVID-19 pandemic began. However, it is important to note a decrease in accidents since 2016, slightly reducing accidents in the following years.

Figure 1. Victims injured in 2021 and 2022 in the city of Bogotá D.C.

Figure 2. Victims who died in 2021 and 2022 in the city of Bogotá D.C.

Figure 1 and 2 show the number of accidents for the year 2021 and 2022 in Bogotá, with 2022 being the year where mobility accidents increased considerably, showing a difference of 41.48% for the injured and 18.18% for the deceased. This shows that the capital handles a constant problem of accidents in which most people are injured or in the worst case, deceased, that is why the purpose of the article is to predict the number of accidents in the city of Bogotá D.C in all localities in order to establish the most common causes that occur in the accidents to determine concrete solutions. applicable to each locality.

And for this we want to implement predictive analysis to find the causes that allow us to find a solution to this problem in Bogotá, initially a supervised learning algorithm called Random Forest will be used, focused on regression, which creates several decision trees and are combined to obtain a more accurate prediction, where, As the model is random, the best characteristic is sought among a subset of characteristics which will generate diversity in a highly stable and secure model.[8] [9]

2. Materials and Methods

2.1 Study Design

Below are the different materials used for the development of the prediction algorithm:

First, this file was handled which was obtained from the Open Data of Bogotá presented in , which contains the structure shown in figure 3.[10]

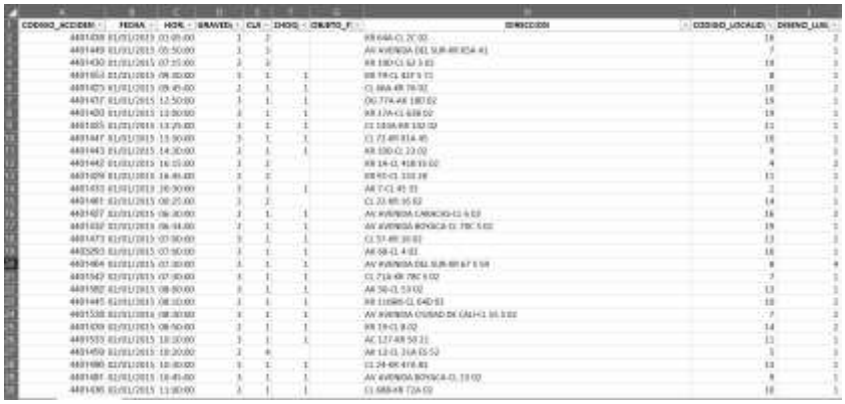


Figure 3. Data presented on accidents in Bogotá D.C since 2015.

As shown in Figure 3, a dataset with 196,153 records is presented that shows a large amount of information about the claims, such as: code, date, severity, and the code of the locality, among other data, which will be taken for their respective analysis and in this way determine the causes of the accidents in a respective locality. which has the following numbers (See table 2).

Table 2. Locations Stipulated in the Claims Document

Locality	Number
Usaquén	1
Chapinero	2
Santa Fe	3
San Cristobal	4
Usme	5
Tunjuelito	6
Bosa	7
Kennedy	8
Fontibón	9
Engativá	10
Suba	11
United Neighborhoods	12
Teusaquillo	13
The Martyrs	14
Antonio Nariño	15
Aranda Bridge	16

Candlemas	17
Rafael Uribe Uribe	18
Bolivar City	19
Sumapaz	20

Figure 4 shows another set of data dating the causes of the accident generated, with a cause code and date, which will be joined to the main data set so that they can be analyzed together and thus be able to predict the causes of the accident.

	A	B	C
1	CODIGO_ACCIDENTE	FECHA	CODIGO_CAUSA
2	4401425	01/01/2015	115
3	4401437	01/01/2015	104
4	4401453	01/01/2015	157
5	4401423	01/01/2015	112
6	4401430	01/01/2015	157
7	4401425	01/01/2015	139
8	4401429	01/01/2015	110
9	4401433	01/01/2015	121
10	4401438	01/01/2015	157
11	4401437	01/01/2015	127
12	4401442	01/01/2015	116
13	4401442	01/01/2015	411

Figure 4. Casualty scenario data from the first data set

Finally, since the prediction algorithm will be created in the Python programming language, Figure 5 shows the space provided by Google called "Google Colab" in which it is allowed to program with this language, also, three libraries imported from Python are shown that will make it possible to work and manage information in a simpler way, to be able to join the data, leave the most important and in this way train and test the model that is going to be generated from the learning, these libraries are Pandas, Numpy and Scikit-Learn. It is important to mention that Sklearn imports algorithm training and Random Regression Forest.

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.model_selection import train_test_split
4 from sklearn.ensemble import RandomForestRegressor
```

Figure 5. Google Colab programming space with imported libraries

2.2 Preparation of Information

The file information was separated and manually deleted in an Excel file before being imported into the workspace since it presented several sheets that were not relevant to the purpose of

the project. Only the two tables mentioned above were separated while the others were eliminated because they did not provide relevant information for the implementation of the algorithm.

2.3 Data Processing

Since not all the information presented is relevant to the algorithm, the data was processed so that it contained only the necessary data. For this, the Pandas and Numpy libraries were used. Pandas allows you to manipulate data through various structures and operations for future analysis; It presents the functions to read, restructure, add, delete, convert, mix and join the data that is used. In addition to offering statistics, regressions, etc. On the other hand, Numpy allows you to create matrices and vectors in several dimensions with great functions focused on high-level mathematics for manipulation. [9]

With the use of both libraries it is allowed to concatenate the tables and eliminate the information globally, then, the data of Severity, Shock, Codigo_Localidad, Diseño_Lugar and Codigo_Causa are saved, likewise, so that the data did not contain processing errors with the predictive algorithm, null data were eliminated and all the values were converted from string to integers in order that the algorithm will not fail and will accept all the information in the table.

The data were predicted using the skicit-learn library, which is a machine learning tool that handles various classification, regression and clustering algorithms using various algebra functions and operations between vectors. This library analyzes the data that is the training information and then generates the test data that is completely randomized, this is done to analyze all the values "X" of training and testing with their respective output "And", so that then the algorithm that, as mentioned above is the Random Forest, processes that information and predicts the respective values for "And" that should approximate the real value. [11]

Results

For prediction, the data was first imported and prepared, saving it in a Pandas DataFrame (Figure 6 and 7).

```
import pandas as pd
df1 = pd.read_excel("siniestros.xlsx") #LEER ARCHIVO
df1.head(8) # IMPRIME TODAS LAS COLUMNAS Y LAS PRIMERAS 8 FILAS
```

	CODIGO_ACCIDENTE	FECHA	HORA	GRAVEDAD	CLASE	CHOQUE	OBJETO_FIJO	DIRECCION	CODIGO_LOCALIDAD	DISENO_LUGAR
0	4401438	01/01/2015	01:05:00	2	2	NaN	NaN	KR 64A-CL 2C 02	16	2
1	4401449	01/01/2015	05:50:00	2	3	NaN	NaN	AV AVENIDA DEL SUR-KR 85A 41	7	1
2	4401430	01/01/2015	07:15:00	2	3	NaN	NaN	KR 18D-CL 62 S 02	19	1
3	4401453	01/01/2015	09:30:00	3	1	1.0	NaN	KR 78-CL 42F S 71	8	1
4	4401423	01/01/2015	09:45:00	2	1	1.0	NaN	CL 66A-KR 76 02	10	2
5	4401437	01/01/2015	12:50:00	3	1	1.0	NaN	DG 77A-AK 18D 02	19	1
6	4401420	01/01/2015	13:00:00	3	1	1.0	NaN	KR 17A-CL 63B 02	19	1
7	4401425	01/01/2015	13:25:00	3	1	1.0	NaN	CL 143A-KR 142 02	11	1

Figure 6. Google Colab space with the first data containing the data presented on accidents in Bogotá D.C since 2015.

```
import pandas as pd
df2 = pd.read_excel("hipotesis.xlsx")
df2.head(8)
```

CODIGO_ACCIDENTE	FECHA	CODIGO_CAUSA
4401425	01/01/2015	115
4401437	01/01/2015	104
4401453	01/01/2015	157
4401423	01/01/2015	112
4401430	01/01/2015	157
4401425	01/01/2015	139
4401429	01/01/2015	116
4401433	01/01/2015	121

Figure 7. Google Colab space with the second data containing the data of the hypotheses of the accidents.

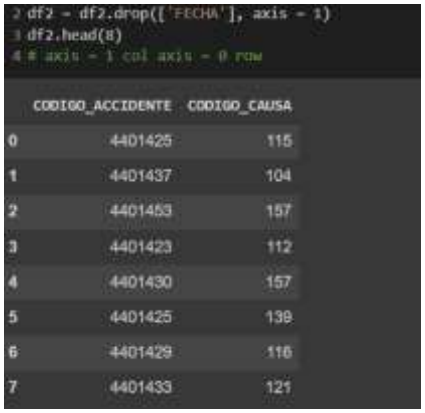
It is observed that the data are presented the same as shown in figures 3 and 4 with all their respective rows and columns in both tables, then proceed to eliminate the Time, Address, Objeto_Fijo, Class and Date , since they are not relevant when generating the prediction to the Code of Cause through the drop parameter of the Pandas library (Figure 8).

```
df1 = df1.drop(['HORA', 'DIRECCION', 'OBJETO_FIJO', 'CLASE','FECHA'], axis = 1) # axis = 1 col axis = 0 row
df1.head(8)
```

	CODIGO_ACCIDENTE	GRAVEDAD	CHOQUE	CODIGO_LOCALIDAD	DISENO_LUGAR
0	4401438	2	NaN	16	2
1	4401449	2	NaN	7	1
2	4401430	2	NaN	19	1
3	4401453	3	1.0	8	1
4	4401423	2	1.0	10	2
5	4401437	3	1.0	19	1
6	4401420	3	1.0	19	1
7	4401425	3	1.0	11	1

Figure 8. Google Colab space with the first modified data showing the relevant data for the algorithm.

This same procedure is performed in the second table, but only with the Date value (Figure 9).

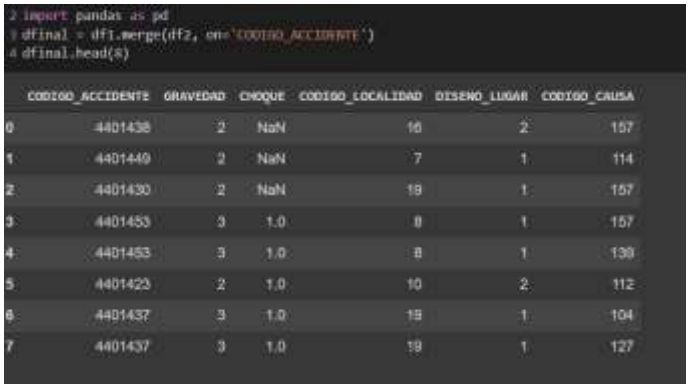


```
df2 = df2.drop(['FECHA'], axis = 1)
df2.head(8)
# axis = 1 col axis = 0 row
```

	CODIGO_ACCIDENTE	CODIGO_CAUSA
0	4401425	115
1	4401437	104
2	4401453	157
3	4401423	112
4	4401430	157
5	4401425	139
6	4401429	116
7	4401433	121

Figure 9. Google Colab space with the second modified data showing the data relevant to the algorithm.

Next, both tables are concatenated through the use of the panda function "**merge**", it is sought to concatenate the second table with the first by means of the common value that is the Accident Code, in this way, it will be possible to know the cause code and the conditions with which it happened (Figure 10).



```
import pandas as pd
dfinal = df1.merge(df2, on='CODIGO_ACCIDENTE')
dfinal.head(8)
```

	CODIGO_ACCIDENTE	GRAVEDAD	CHOQUE	CODIGO_LOCALIDAD	DISENO_LUGAR	CODIGO_CAUSA
0	4401438	2	NaN	16	2	157
1	4401449	2	NaN	7	1	114
2	4401430	2	NaN	19	1	157
3	4401453	3	1.0	8	1	157
4	4401453	3	1.0	8	1	139
5	4401423	2	1.0	10	2	112
6	4401437	3	1.0	19	1	104
7	4401437	3	1.0	19	1	127

Figure 10. Google Colab space with both tables concatenated.

After joining both tables, we proceeded to eliminate the null values (NaN) of the table in the Shock section using the "dropna" function, since it was observed that several values there presented the empty data and prevented the algorithm from working correctly (Figure 11).

```
dfinal.dropna(subset=['CHOQUE'], inplace= True)
dfinal.head(8)
```

	CODIGO_ACCIDENTE	GRAVEDAD	CHOQUE	CODIGO_LOCALIDAD	DISENO_LUGAR	CODIGO_CAUSA
3	4401453	3	1.0	8	1	157
4	4401453	3	1.0	8	1	139
5	4401423	2	1.0	10	2	112
6	4401437	3	1.0	19	1	104
7	4401437	3	1.0	19	1	127
8	4401420	3	1.0	19	1	103
9	4401425	3	1.0	11	1	115
10	4401425	3	1.0	11	1	139

Figure 11. Google Colab space with table without Shock null values

Also, when concatenating both tables, some values are automatically duplicated, therefore, the duplicates are eliminated with the parameter "drop_duplicates" where it is specified that the last value is preserved if it is repeated (Figure 12).

```
dfinal = dfinal.drop_duplicates(subset=['CODIGO_ACCIDENTE'], keep='last')
dfinal.head(20)
```

	CODIGO_ACCIDENTE	GRAVEDAD	CHOQUE	CODIGO_LOCALIDAD	DISENO_LUGAR	CODIGO_CAUSA
4	4401453	3	1.0	8	1	139
5	4401423	2	1.0	10	2	112
7	4401437	3	1.0	19	1	127
8	4401420	3	1.0	19	1	103
10	4401425	3	1.0	11	1	139
11	4401447	3	1.0	10	1	121
12	4401443	3	1.0	9	1	103

Figure 12. Google Colab space with the table without the duplicate values.

As the tables are concatenated it is possible to delete the Accident Code, so proceed to eliminate it using the "drop" function (Figure 13).

```
dfinal = dfinal.drop(['CODIGO_ACCIDENTE'], axis = 1)
dfinal.head(8)
```

	GRAVEDAD	CHOQUE	CODIGO_LOCALIDAD	DISEÑO_LUGAR	CODIGO_CAUSA
4	3	1.0	8	1	139
5	2	1.0	10	2	112
7	3	1.0	19	1	127
8	3	1.0	19	1	103
10	3	1.0	11	1	139
11	3	1.0	10	1	121
12	3	1.0	9	1	103
16	3	1.0	2	1	121

Figure 13. Google Colab space with the table without the accident code.

To ensure data integrity, the cause code, the shock, the locality code are converted to integer value using the "astype" function, specifying "int" (Figure 14).

```
dfinal = dfinal[pd.to_numeric(dfinal['CODIGO_CAUSA'], errors='coerce').notnull()]

dfinal['CHOQUE'] = dfinal['CHOQUE'].astype('int')
dfinal['CODIGO_LOCALIDAD'] = dfinal['CODIGO_LOCALIDAD'].astype('int')
dfinal['CODIGO_CAUSA'] = dfinal['CODIGO_CAUSA'].astype('int')
```

Figure 14. Google Colab space with the code to change the specified values to integer.

Once completed, you get the following table with some of the total values:

Table 3. Final table with which the prediction algorithm will be worked

Item	Gravity	Crash	Codigo_Localidad	Diseño_Lugar	Codigo_Causa
4	3	1	8	1	139
5	2	1	10	2	112
7	3	1	19	1	127
8	3	1	19	1	103
10	3	1	11	1	139
11	3	1	10	1	121
12	3	1	9	1	103
16	3	1	2	1	121

Once this table is obtained, the values of "X" and "Y" are set. To select the value of "Y" the function "iloc" is used. For the variable

"X" the first four columns of the table are defined which are Severity, Shock, Codigo_Localidad and Diseño_Lugar, this is information that determines the cause of the accident.

On the other hand, for the variable "Y" only the Cause Code is defined, since it is the value that is sought to be found from the other parameters (Figure 15).

```
X = dfinal.iloc[:, 0:4].values
y = dfinal.iloc[:, 4].values
```

Figure 15. Google Colab space with X and Y variables defined.

After the values and variables to work have been chosen, we proceed with the training method, for this the method of the "skicit-learn" library is imported and four variables are defined, two of them are the original data and the other two are values of tests obtained at random, precisely so that the algorithm is trained. It was decided to conduct a training test of 30% of the data, that is, taking approximately 50,270 values.

```
#Separar la data en entrenamiento y test
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)
```

Figure 16. Google Colab space with the code to separate the data into training and testing.

When starting the training method, we proceed to start with the algorithm. The working method is imported, which in this case is Random Forest Regression; Once the training data is structured within the tree, another variable is created which will seek to predict based on the variable "Y" taking into account the "X" test. In this way he will apply the Random Forest method in order to predict the values and be able to generate a cause code with the characteristics of "X" (Figure 17).

```
from sklearn.ensemble import RandomForestRegressor
algoritmo = RandomForestRegressor(n_estimators = 2, random_state = 0)
algoritmo.fit(X_train, y_train)
y_pred = algoritmo.predict(X_test)
```

Figure 17. Google Colab space with the code to apply the Random Forest Regressor training method.

Finally, once the algorithm is completed, the values of the "Y test" are compared with the "Predicted Y" to observe the changes and accuracy of the model (Figure 18).

```
dcomp = pd.DataFrame({"Valor de Prueba": y_test,
                      "Valor Predicho": y_pred,
                     })
```

Figure 18. Google Colab space with the comparison table between the values of "And test" and "And predicted".

Table 4 shows a summary of the test values versus the values predicted by the algorithm.

Table 4. Comparison of test values against predicted values

Index	Test Value	Predicted Value
0	121	131
1	103	124
2	104	121
3	157	126
4	142	117
...	...	...
995	157	126
996	121	123
997	306	128
998	121	122
999	105	128

When measuring the accuracy of the Random Forest Regression model, it can be seen that an accuracy of 87.66% is obtained (Figure 19).

```
y_pred = y_pred.astype(int)
form = (abs(y_pred-y_test)/y_test)*100
prec = 100 - np.mean(form)
print("Precision del modelo: %.2f"%prec,"%")
```

```
Precision del modelo: 87.66 %
```

Figure 19. Google Colab space with the code and the result of the percentage of accuracy of the model.

Conclusions

The implementation of the Random Forest Regression algorithm allows obtaining an accuracy of 87.66%, which is an acceptable percentage in predictive modeling and gives a high degree of reliability that these are probably the causes of accidents at the time that an accident may occur.

These causes were obtained so that there can be another study that makes it possible to determine direct solutions to the road conditions found in Bogotá, with this, it is possible to give an idea of how to manage the potential causes if they are constantly repeated in a locality and that will help the well-being of the community in general and specifically in the road aspect, Finally, it will be possible to reduce the accidents that each year claim fatalities.

Finally, artificial intelligence has proven to be a useful tool for predicting traffic accidents, providing a preventive approach that significantly improves road safety. And, on the other hand, help to predict and mitigate traffic accidents, optimizing response times to such emergencies and helping the design of public policies focused on prevention.

References

[1] W. Eckerson, «Predictive Analytics,» RESEARCH & RESOURCES, 10 5 2007. [Online]. Available: https://tdwi.org/articles/2007/05/10/predictive-analytics.aspx?sc_lang=en. [Accessed 28 07 2023].

[2] L. J. Sandoval, «MACHINE LEARNING ALGORITHMS FOR DATA ANALYSIS AND PREDICTION,» REVISTA TECNOLÓGICA, vol. 11, nº 1, pp. 36-40, 2018.

[3] DECIDE4AI, «Decide4AI,» Decide4AI, 10 07 2020. [Online]. Available: https://decidesoluciones.es/ejemplos-implementacion-analitica-predictiva-machine-learning/. [Accessed 28 07 2023].

[4] C. Espino Timón, Final degree project: Predictive analysis: techniques and models used and applications of the same - Open Source tools that allow its use, Catalonia: Open University of Catalonia, 2017.

[5] J. Cortes Méndez, A. Abuchar Porras, F. Blanco Garrido, J. A. Paez, J. Palacios Rozo and F. A. Simanca H., «Algorithm for predicting the most frequent causes of mortality by analyzing age and gender variables,» Journal of Positive Psychology and Wellbeing, vol. 6, nº 1, pp. 1419-1429, 2022.

[6] F. A. Simanca H., F. Blanco Garrido, J. Lemus, W. Torres Cerón, P. E. Carreño H., J. Palacios Rozo and L. Barbosa Guerrero, «Air Quality Index Prediction Model for the City of Bogotá, D.C.,» Advances in Mechanics, vol. 9, nº 3, pp. 542-553, 2021.

[7] L. Barbosa Guerrero, F. A. Simanca H., M. Hernández Bejarano, A. Alfaro Tejeiro, J. Palacios Rozo, F. Blanco Garrido and C. Ortíz Daza, «Application of the Polynomial Regression Algorithm to Predict Covid-19 Cases Per Day in Colombia,» Advances in Mechanics, vol. 9, nº 3, pp. 49-61, 2021.

[8] C. Gómez Guasca, «17% of companies apply data analysis to innovate,» Portfolio, pp. https://www.portafolio.co/negocios/empresas/empresas-en-colombia-y-uso-de-datos-para-innovar-procesos-555891, 4 10 2021.

[9] Road Safety Agency, "Road accidents cost the country 23.9 billion pesos a year," Portfolio, pp. https://www.portafolio.co/economia/a-octubre-en-colombia-fallecieron-4-156-personas-en-siniestros-viales-546657, 15 11 2020.

[10] National Road Safety Agency, "National Road Safety Agency," 16 04 2022. [Online]. Available: https://ansv.gov.co/es/observatorio/estad%C3%ADsticas/historico-victimas. [Accessed 07 28 2023].

[11] E. Russano and E. Ferreira Avelino, FUNDAMENTALS OF MACHINE LEARNING USING PYTHON, Oakville: Arcler Press, 2020.

[12] J. Gironés Roig, J. Casas Roma, J. Minguillón Alfonso and R. Caihuelas Quiles, Data mining: Models and algorithms, Catalonia: Editorial UOC, 2017.

[13] District Secretary for Mobility, "Open Data," District Secretary for Mobility, 20 10 2021. [Online]. Available: https://datosabiertos.bogota.gov.co/dataset/0b070626-fe5a-42dc-ae99-92601da166d9?_external=True. [Accessed 10 07 2023].

[14] D. Jared, Big Data, Data Mining and Machine Learning, New Jersey: Wiley, 2014.