FINAL PRACTICE

The tools of the data scientist

Brief description

This Exploratory Data Analysis (EDA) exercise examines traffic accident data in Madrid from 2019 to 2024 using key variables such as date, time, location, type of accident, weather conditions, and characteristics of those involved (age, gender, vehicle). The goal is to explore, clean, and visualize trends.

Sergio Hervás Aragón

Sergiohervas9@gmail.com

For this exploratory analysis (EDA), we will work with an open data portal (Open Data) to ensure that the data is real and reliable.

On this occasion, we will use data on **"Traffic Accidents in the City of Madrid,"** which has been collected and recorded by the Madrid Municipal Police. All the dataset information can be found at the following link:

Traffic Accidents in the City of Madrid

The information included in each file is recorded per person involved in an accident. Therefore, a single accident may have more than one person involved.

Looking at the historical data series, records have been collected since 2010. In 2019, there was a change in how the information is gathered, resulting in different column structures. To streamline the file concatenation process, we will work exclusively with data from **2019 to 2023** (inclusive).

As an optional step, to get familiar with the dataset, the Open Data portal also provides visualizations related to traffic accidents.
Visualization of Traffic Accidents in Madrid

Variable Information

Each dataset must include the following columns:

- **num_expediente**: Accident identifier.
- **fecha**: Date when the accident occurred.
- **hora**: Time when the accident occurred.
- **localizacion**: Street where the accident occurred.
- **numero**: Street number where the accident occurred.
- **cod_distrito**: Code of the district where the accident occurred.
- **distrito**: Name of the district where the accident occurred.
- **tipo_accidente**: Category describing how the accident occurred.
- **estado_meteorológico**: Weather conditions at the time of the accident.
- **tipo_vehiculo**: Category referring to the type of vehicle involved in the accident.
- **rango_edad**: Age range of the person involved.
- **sexo**: Gender of the person involved.
- **cod_lesividad**: Code indicating the severity of the injuries of the person involved, referring to whether they required medical assistance.
- **lesividad**: Type of medical intervention provided to the person involved.
- **coordenada_x_utm**: X coordinate.
- **coordenada_y_utm**: Y coordinate.
- **positiva_alcohol**: Indicates whether an alcohol test was conducted and whether the person tested positive.
- **positiva_droga**: Indicates whether a drug test was conducted and whether the person tested positive.

OBJECTIVES

With the given datasets and variables given, you are required to:

1. **Load all CSV files into a single Dataframe.**
   How many total rows do you obtain?

2. **Remove all columns related to coordinates,** as we won't be using them.

3. **Restructure the vehicle type column** to reduce redundant information and include only the following categories:

   - **Car (Turismo)**
   - **Motorcycle (Motocicleta)**
   - **Van (Furgoneta)**
   - **Bicycle (Bicicleta)**
   - **Truck (Camión)**
   - **Bus (Autobús)**
   - **Other Vehicle (Otro vehículo)**

You are free to adjust the restructuring as needed. Below is an example of how to group different vehicle types:

Motorcycle (Motocicleta) should include:

- Motorcycle up to 125 cc
- Motorcycle > 125 cc
- Moped
- Three-wheel motorcycle > 125 cc
- Two-wheel moped (L1e-B)
- Three-wheel motorcycle up to 125 cc
- Motorized cycle (L1e-A)

Bicycle (Bicicleta) should include:

- Bicycle
- EPAC bicycle (pedal-assisted)
- Cycle

Bus (Autobús) should include:

- Bus
- Articulated bus
- EMT bus
- Articulated EMT bus
- Minibus (up to 17 seats)

Truck (Camión) should include:

- Rigid truck
- Tractor-truck
- Etc.

Finally, display the number of new categories along with their frequency.

NOTE: We will also transform the null values in this column to null.

4. **Let's analyze null values:**
   Inspect the null values in all columns.
   Check if there are any columns that consist entirely of null values. If so, delete those columns.

   - For the **"positive_drug"** column, fill the null values with 0.
   - For the **"positive_alcohol"** column, fill the null values with "N".
   - For the columns related to **injury severity**, fill the missing data with "No medical attention" (in the **"injury_code"** column, use the value 0).
   - For the **"weather condition"** column, use the existing category "Unknown".
   - Remove the remaining null values from the dataset.

   How many rows and columns do you have now?

5. **Let's continue reducing the number of categories in the categorical columns:**
   In this case, we will create a new category called **"Other Accident"** for all categories in the **"accident_type"** column that account for less than 10% of the total.

6. **Regarding the severity of traffic accidents, as we are analyzing, there are drivers who tested positive for alcohol and others who tested positive for drug use. But is there any accident where those involved tested positive for both alcohol and drugs?**

   Display the number of individuals involved, as well as the number of different case files.

7. **What is the most common type of accident for those involved who tested positive for alcohol? And for those who did not test positively for alcohol? What differences do you observe?**
8. **For each type of vehicle, visually display the number of accidents based on the weather condition.**
   NOTE: In this section, treat each row as an accident, meaning you don't need to group by case number.
9. **Group the dataframe by case number, and let's analyze whether there are multiple accidents. From the previous grouping, obtain all case numbers that involve five or more distinct types of vehicles.**

   - How many case numbers appear?

   - How many individuals are involved in each case?

   - What different types of vehicles appear in each case number?
     TIP: Investigate the filter function and nunique() for a column.

10. **Take the "hour" column and extract only the hour (for example, from 9:10:00, extract 09). After that, graphically show which hours are the most dangerous for driving in Madrid.**