Ejercicio 1 – EDA (Exploratory Data Analysis) Python

Para este análisis exploratorio (EDA), vamos a trabajar desde un portal de datos abiertos (Open Data), ya que podemos garantizar de esta manera que los datos son reales.

En esta ocasión, trabajaremos con datos sobre "Accidentes de tráfico en la Ciudad de Madrid", todos estos accidentes son recopilados y registrados por la Policía Municipal de Madrid Toda la información del dataset podemos encontrarla en el siguiente enlace:

Accidentes de tráfico de la Ciudad de Madrid

Como tal, la información que se incluye en cada archivo es un registro por persona implicada en un accidente, por lo tanto, un accidente puede tener más de una persona implicada.

Si observamos la serie histórica de los datos, se llevan recopilando desde 2010, en el año 2019 se realiza un cambio en la forma de obtener la información por lo que las columnas son diferentes, para favorecer las tareas de concatenación de archivos trabajaremos únicamente con los años. **Desde 2019 hasta 2023** (ambos años incluidos).

De forma opcional, para ir familiriazándose con el cojunto de datos, desde este portal Open Data, también existen visualizaciones realizadas sobre accidentes de tráfico. Visualización sobre accidentes de tráfico en Madrid

Información de las variables

Cada dataset tendrá que contener las siguientes columnas:

- **num_expediente**: Identificador del accidente.
- **fecha**: Fecha en la que se produce el accidente.
- **hora**: Hora en la que se produce el accidente.
- **localizacion**: Calle en la que se produce el accidente.
- **numero**: Número de la calle en la que se produce el accidente.
- **cod_distrito**: Código del distrito en el que se produce el accidente.
- **distrito**: Nombre del distrito en el que se produce el accidente.
- **tipo_accidente**: Categoría que registra cómo se produjo el accidente.
- **estado_meteorológico**: Estado climatológico de la vía cuando se produjo el accidente.
- **tipo_vehiculo**: Categoría referida al tipo de vehículo implicado en el accidente.
- **rango_edad**: Rango de edad de la persona implicada.
- **sexo**: Sexo de la persona implicada.
- **cod_lesividad**: Código referido a la lesividad del implicado, se refiere a si tuvo algún tipo de asistencia médica.
- **lesividad**: Tipología de la intervención realizada a la persona implicada.
- **coordenada_x_utm**. Coordenada X
- **coordenada_y_utm**: Coordenada Y
- **positiva_alcohol**: Si se realiza la prueba de alcholemia, si la persona implicada da positivo o no.
- **positiva_droga**: Si se realiza el test de drogas, si la persona implicada da positivo o no.

OBJETIVOS

Teniendo estos datasets y sus variables se pide:

1. Carga todos los csv en un único dataframe ¿Cuántas filas totales obtienes?
2. No vamos a emplear variables que estén referidas a coordenadas, borra todas aquellas columnas que estén referidas a coordenadas.
3. Si observamos la columna que contiene el tipo del vehículo, encontramos mucha información redundante, vamos a reestructurar esta columna para tener solamente las siguientes categorías:
    1. Turismo
    2. Motocicleta
    3. Furgoneta
    4. Bicicleta
    5. Camión
    6. Autobús
    7. Otro vehículo.

En este sentido siéntete libre de realizar tu propia reestructuración, este, sería un ejemplo:

- Crear la categoría **Motocicleta** que contenga:
- Motocicleta hasta 125 cc
- Motocicleta > 125cc
- Ciclomotor
- Moto de tres ruedas > 125cc
- Ciclomotor de dos ruedas L1e-B
- Moto de tres ruedas hasta 125cc
- Ciclo de motor L1e-A
- **Bicicleta** que contenga:
- Bicicleta
- Bicicleta EPAC (pedaleo asistido)
- Ciclo
- **Autobús** que contenga:
- Autobús
- Autobús articulado
- Autobús EMT
- Autobús articulado EMT
- Microbús <= 17 plazas
- **Camión** que contenga:
- Camión rígido
- Tractocamión
- Etc.

Finalmente, muestra el nº de categorías nuevas junto con su frecuencia

NOTA: Los valores nulos de esta columna los transformaremos también en nulos.

4. Vamos a analizar valores nulos, inspecciona los valores nulos de todas las columnas.
    1. Revisa si hay columnas que son íntegramentevalores nulos, si esto ocurre, borra esas columnas
    2. Para la columna positivo droga, rellena losnulos con 0.
    3. Para la columna positivo alcohol, rellena losnulos con “N”
    4. Para las columnas referidas a lesividad,rellenaremos los datos faltantes con “Sin atención sanitaria” (en códigolesividad, pondremos valor 0)
    5. Para el estado meteorológico emplearemos lacategoría ya existente “Se desconoce”
    6. El resto de valores nulos, los eliminaremos delconjunto de datos.
    7. ¿Cuántas filas y columnas tienes ahora?

5. Vamos a seguir reduciendo el número de categorías de las columnas categóricas, en este caso, vamos a generar una nueva categoría llamada “Otro accidente” para todas aquellas categorías que tengan un porcentaje inferior al 10% en lacolumna tipo accidente.
6. Dentro de la gravedad de un accidente de tráfico como estamos comprobando hay conductores que dieron positivo por alcohol y, otros positivos por consumo de droga, pero ¿Hay algún accidente en donde los implicados dieran positivo en la prueba de alcohol y también en la prueba de drogas? Muestra el número de implicados, así como el número de expedientes diferentes.
7. ¿Cuál es el tipo de accidente más común para aquellos implicados que habían dado positivo en alcohol? ¿y para aquellos implicados que no dieron positivo en la prueba de alcohol? ¿Qué diferencias observas?
8. Para cada tipo de vehículo muestra visualmente el número de accidentes en funcióndel estado meteorológico.
NOTA: En este apartado tomaremos cada fila como unaccidente, es decir, no tienes porqué agrupar por número de expediente.

9. Agrupa el dataframe por el número de expediente, vamos a analizar si hay accidentes múltiples. De la agrupación anterior obtén todos aquellos números de expediente que tengan involucrados mayor o igual que 5 tipos distintos de vehículos.

1. ¿Cuántos números de expediente aparecen?
2. ¿Qué cantidad de implicados hay en cadaexpediente?
3. ¿Qué tipos de vehículos diferentes aparecen encada número de expediente?

PISTA: Investigalas funciones filter y para una columna nunique()

- 10. Toma la columna hora y, quédate solamente la hora, es decir de 9:10:00 solo obtener 09, tras ello, muestra gráficamente cuáles son las horas más peligrosas para circular en Madrid