Contenido de la actividad

Top of Form

**GRUPO 1**

**Práctica final módulo: Las herramientas del científico de datos**

**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](https://datos.madrid.es/portal/site/egob/menuitem.c05c1f754a33a9fbe4b2e4b284f1a5a0/?vgnextoid=7c2843010d9c3610VgnVCM2000001f4a900aRCRD&vgnextchannel=374512b9ace9f310VgnVCM100000171f5a0aRCRD&vgnextfmt=default)

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](https://visualizadatos.madrid.es/pages/accidentes-de-trafico)

**Información de las variables**

Cada dataset tendrá que contener la 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:

* + Carga todos los csv en un único dataframe ¿Cuántas filas totales obtienes?
  + No vamos a emplear variables que estén referidas a coordenadas, borra todas aquellas columnas que estén referidas a coordenadas.
  + 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:
  + Turismo
  + Motocicleta
  + Furgoneta
  + Bicicleta
  + Camión
  + Autobús
  + 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.
  + Revisa si hay columnas que son íntegramentevalores nulos, si esto ocurre, borra esas columnas
  + Para la columna positivo droga, rellena los nulos con 0.
  + Para la columna positivo alcohol, rellena los nulos con “N”
  + Para las columnas referidas a lesividad,rellenaremos los datos faltantes con “Sin atención sanitaria” (en códigolesividad, pondremos valor 0)
  + Para el estado meteorológico emplearemos lacategoría ya existente “Se desconoce”
  + El resto de valores nulos, los eliminaremos delconjunto de datos.
  + ¿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.
  + ¿Cuántos números de expediente aparecen?
  + ¿Qué cantidad de implicados hay en cadaexpediente?
  + ¿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

**Ejercicio 2 – EDA (Exploratory Data Analysis) R**

En este caso práctico, vamos a realizar un Análisis Exploratorio sobre los precios del mercado energético de Europa desde 2010. La fuente original de los datos puede encontrarse en este enlace https://zenodo.org/record/5977857#.Y-pQT3bMKUk aunque, tomaremos como referencia su publicación en Kaggle.

* + https://www.kaggle.com/datasets/thedevastator/hourly-european-power-market-prices

**Información de las variables**

Este conjunto de datos contiene precios de energía de mercado diarios europeos por hora de varios sistemas y tipos de monedas. En general, el conjunto de los datos ofrece la siguiente información:

* + **fecha**: Fecha en la que se genera el dato de transacción.
  + **hora**: Hora en la que se registra la transacción.
  + **sistema**: Mercado Europeo regulador de los precios de energía
  + **bandera**: Si es horario invierno o verano
  + **precio**: Precio que marca el sistema para el precio de la energía.
  + **tipo\_moneda**: Tipo de moneda, pudiendo ser euro o libra
  + **origen\_dato**: Web desde la que se ingesta el registro del precio.
  + **fecha\_actualizacion**: Última fecha de verificación del dato.

**OBJETIVOS**

Teniendo en cuenta el siguiente conjunto de datos, se pide hacer lo siguiente:

* + Carga el dataset como dataframe y, borra si hay alguna columna que no aporte información como IDs.
  + Obtén el rango de las fechas ¿Para cuántos años tenemos registros de los datos?
  + Sin importar el año, ¿Qué día del mes se reportan más registros? ¿Cuál es el que menos?
  + Si analizamos la variable precio, parece que hay precios del mercado energético que aparecen en negativo, toma todos aquellos valores que sean negativos y, transformalos en nulo ¿Qué cantidad hay de valores nulos?. Posteriormente realiza estas dos operaciones:
  + 4.1 Asigna un cero a todos los valores nulos menos a los que pertenezcan al mercado de Dinamarca (DK1 y DK2), puedes consultar más información sobre los nombres de los mercados aquí: https://www.jao.eu/sites/default/files/2022-03/List%20of%20Bidding%20Zone%20borders\_2022\_CORE.pdf
  + 4.2 Borra el resto de nulos.
  + ¿Cuántos datos había antes de borrar nulos y, cuantos quedan ahora?, es decir ¿Cuántos datos se han borrado?
  + La variable bandera presenta los siguientes valores 0 y 1, estos valores están referidos al horario de invierno 0 y horario de verano 1, crea una nueva variable categórica llamada ESTACION que tenga los valores invierno y verano. Investiga la función recode del paquete dplyr
  + La columna tipo de moneda tiene dos valores 1 que es €/MWh y 2 GBP/MWh, recodifica la propia variable a los valores categóricos Euros y Libras. Por lo tanto, tenemos precios en dos monedas diferentes, analiza el precio medio del mercado energético en función de la estación y el tipo de moneda.
  + Toma la columnas fecha\_actualizacion y, toma únicamente la parte de la fecha sin contar la hora y, obtén la diferencia en días entre la fecha de actualización y la columna fecha. Tras obtener dicha diferencia muestra el promedio de días de actualización, cuanto más alto mejor, eso querrá decir que datos muy antiguos siguen teniendo registros de actualización: Pista, investiga la función difftime y as.Date
  + Filtra el dataset por el tipo de moneda Euros y el mercado español, visualiza la serie completa respecto a la fecha de registro del precio del mercado energético. Actualmente, vivimos en tiempos de bastante inestabilidad en el mercado energético ¿se ve reflejado en la visualización?
  + Toma el precio medio en euros por cada año del mercado energético de Francia y España. Visualiza la evolución del precio medio de ambos mercados
  + Sin importar la moneda, muestra un top 10 para los países que tengan el precio medio más alto en el mercado energético y, el top 10 con el valor más pequeño.

##### **Condiciones de entrega**

En el campus (Blackboard) en la sección indicada para la práctica final, se hará entrega de un archivo comprimido .rar o .zip que presente el siguiente formato APELLIDO1\_APELLIDO2\_NOMBRE.rar, dentro del archivo comprimido se incluirán ambos notebooks (uno en formato .ipynb y otro en formato .RMD), además se deberán enviar las versiones en HTML de cada uno de los notebooks.

No es necesario realizar ninguna memoria u otro documento explicativo sobre los ejercicios, todos los comentarios o argumentaciones sobre las prácticas se realizarán utilizando Markdown en cada notebook.

No es necesario incluir los dataset de entrada en el archivo de entrega.

##### **Información sobre la práctica**

Fecha de plazo límite para su entrega:

-**07/01/2024 - 23:59**

* Cualquier entrega realizada después de dicha fecha límite se considerará suspensa.
* En caso de detectarse plagio la práctica estará suspensa
* La práctica se realiza de forma individual.
* La calificación y observaciones de la práctica se recibirán como respuesta de la que contenga la práctica.
* Las correcciones y calificaciones se irán enviando a partir del 8 de Enero.

PD: Recordad que podéis preguntar dudas jmmoreno@profesorescol.imf.com

##### **Criterios de evaluación**

Evaluación por competencias:

* Desarrollo software en lenguaje R/Python. 30%
* Habilidades de procesamiento de datos (csv, txt, json, etc.) en R/Python. 15%
* Elección, diseño y uso de estructuras de datos en R/Python. 20%
* Empleo de librerías de ciencia de datos (Numpy, Pandas) (Dplyr, Tidyr). 20%
* Competencias en visualización de datos. 15%

La puntuación total de cada ejercicio será de 10 puntos, ambos ejercicios tendrán la misma ponderación, por lo tanto, la puntuación final será la media de la puntuación total de ambos ejercicios. La práctica se considerará aprobada si la puntuación final es superior o igual a 5 puntos. Recordad, la evaluación final cuenta el 80% de la nota

En caso de que algún alumno no supere la puntuación mínima de 5 puntos teniendo en cuenta los ejercicios de Python, R y la práctica final recibirá las pautas pertinentes de recuperación desde el profesorado

Mucha suerte a **todos** !